

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСПІЛКИ
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ»
ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
ФАКУЛЬТЕТ ЕКОНОМІКИ І МЕНЕДЖМЕНТУ
ФОРМА НАВЧАННЯ ДЕННА
КАФЕДРА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТА СОЦІАЛЬНОЇ
ІНФОРМАТИКИ

Допускається до захисту
Завідувач кафедри _____ О.О. Ємець
(підпис)

« _____ » _____ .2020р.

ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО БАКАЛАВСЬКОЇ РОБОТИ

на тему

РОЗРОБКА САЙТУ ІНТЕРНЕТ-МАГАЗИНУ «StoreMobile»

зі спеціальності 122 «Комп'ютерні науки та інформаційні технології»

Виконавець роботи Михайловський Роман Михайлович

_____ «__» _____ 2020 р.
(підпис)

Науковий керівник к.ф.-м.н., доц., Чілікіна Тетяна Василівна

_____ «__» _____ 2020р.

ПОЛТАВА 2020 р.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І	
ТЕРМІНІВ	3
ВСТУП.....	4
1. ПОСТАНОВКА ЗАДАЧІ.....	6
2. ІНФОРМАЦІЙНА ЧАСТИНА.....	9
2.1. Поняття веб-сайта.....	9
2.2. Основні етапи створення веб-сайтів.....	11
2.3. Обґрунтування потреби в веб-сайті.....	14
2.4. Етапи розробки веб-сайта.....	16
2.4.1. Вибір програмних засобів для розробки веб-сайта.....	16
2.4.2. Обґрунтування вибору CMS.....	18
3. ТЕОРЕТИЧНА ЧАСТИНА.....	22
3.1 Розробка веб-дизайну сайту.....	22
3.2 Структура сайту	23
3.3 Наповнення сайту товарами.....	26
4 ПРАКТИЧНА ЧАСТИНА.....	28
4.1 Встановлення на сервер CMS WordPress	28
4.2 Встановлення платного шаблону сайту.....	30
4.3 Встановлення необхідних плагінів на сайт	32
4.4. Особливості налаштування теми	33
4.5. Тестування сайту.....	43
ВИСНОВКИ.....	45
СПИСОК ЛІТЕРАТУРИ.....	46
ДОДАТОК А. Програма.....	48

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

Web	система доступу до пов'язаних між собою документів на різних комп'ютерах, підключених до Інтернету
HTML	HyperText Markup Language — Мова розмітки гіпертекстових документів.
CSS	Cascading Style Sheets (Каскадні таблиці стилів).
CMS	Content Management System (система керування вмістом).
БД	база даних.
FTP	File Transfer Protocol (протокол передачі файлів).
WYSIWYG	What You See Is What You Get (що бачиш, те й отримуєш).
PHP	Hypertext Preprocessor (гіпертекстовий препроцесор).
SEO	search engine optimization (пошукова оптимізація сайту).
XML	eXtensible Markup Language (розширювана мова розмітки).
WP	WordPress (система керування вмістом з відкритим кодом, яка застосовується для створення веб-сайтів).
Back-end	це серверна частина сайту, загрузка інформації, та її опрацювання.
Front-end	це інтерфейс для взаємодії між користувачем і back end.
JS	JavaScript – скриптова мова програмування.
AJAX	Asynchronous JavaScript And XML – підхід до побудови користувацьких інтерфейсів веб-застосунків.

ВСТУП

В умовах масової комп'ютеризації та інформатизії всіх сфер життя та інтеграції України в світовий інформаційний простір важливого значення набуває ефективне використання інформаційно-комунікаційних технологій в усіх сферах життя. Суттєва роль при цьому належить веб-технологіям, які швидко проникають усі сфери суспільства, мета якої забезпечити перехід від індустріального суспільства до інформаційно-технологічного через новаторство у навчанні та науково-методичній роботі.

Глобальна мережа Internet стала невід'ємною частиною життя величезної кількості людей, як у розвинених країнах, так і у таких, що розвиваються. У сучасному житті навіть пересічного учня, студента чи офісного працівника, так і в глобальному інформаційному суспільстві загалом стає неможливим ведення діяльності у будь-якій галузі без доступу до Всесвітньої мережі Інтернет. Завдяки стрімким темпам зростання Інтернет, дедалі частіше у його користувачів постає питання щодо розроблення якщо не власного веб-сайту, то хоча б індивідуальної веб-сторінки.

Створення сайтів на замовлення є прибутковою справою, якою займаються професійно. Разом з тим, на сьогоднішній день існує багато інструментальних засобів створення веб-сайтів, які спрощують це завдання і дають змогу будь-якому користувачеві Інтернету зробити свій сайт власноруч.

З кожним роком Web-сайти стають все більш складними і інтерактивними, а інформація, розміщена на них, – повнішою і якіснішою. У простому випадку сайт є сукупністю статичних документів HTML. Інформація, розміщена на такому сайті, як правило, постійна [6].

Створення веб-сайтів, як правило, є досить складним творчим процесом, що вимагає як технічних знань та навиків, так і креативних зусиль із боку їх творців. Саме це й зумовлює необхідність розглянути веб-сайт не лише як інформаційний ресурс, а ще і як результат інтелектуальної, творчої діяльності та з'ясувати особливості правового режиму веб-сайту як об'єкта права інтелектуальної власності.

Щодня з'являються тисячі нових веб-сайтів, на які може зайти будь-яка людина в цілому світі, при умові, що вона має вихід в Інтернет. Внаслідок цього виникає велика конкуренція між електронними підприємствами, кожне з них намагається збільшити свою цільову аудиторію.

Мотивами для цих дій можуть служити різні причини: отримання прибутку, формування іміджу фірми чи просто дешева популярність, а інколи це просто рекламні заходи, тобто можливість надати певну інформацію (наукову, комерційну, розважальну) широкому загалу користувачів Інтернету з певною метою.

Не має значення, з якої причини був створений сайт, але він має бути кращий за сайт конкурента, тому особливої актуальності набувають питання розробки основних критеріїв для оцінки якості вебсайтів як умови розвитку електронного підприємництва в Україні.

Враховуючи те, що Інтернет на сьогодні в Україні перебуває в процесі свого розвитку і практично не існує великої конкуренції між українськими Інтернет-ресурсами, то досліджувана проблема ускладнюється високим рівнем якості веб-сайтів іноземних компаній.

Таким чином, з нашими сайтами конкурують закордонні веб-сайти, особливо російські, що зумовлено майже відсутністю мовного бар'єру. Хоча наші сайти є конкурентно спроможними, але кількість закордонних інтернет-ресурсів в десятки разів перевищує кількість вітчизняних [13].

Потрібно не тільки збільшувати кількість українських інтернет-ресурсів, але і підвищувати їх якість, як з точки зору привабливості для користувачів, так і відповідно до запропонованої системи критеріїв у даній концепції.

Актуальність бакалаврської роботи полягає в тому, що б розробити сучасний інтернет-магазин під назвою «StoreMobile», який орієнтується на розміщення мобільних пристроїв та додаткових товарів.

Метою роботи є розробка інтернет-магазину, який відповідає технічному завданню отриманому від замовника, а саме мають містити розділи та підрозділи товарів, а також невеликий блог з новинами і можливістю оформити замовлення.

Об'єктом розробки є створення сайту «StoreMobile».

Предмет розробки – веб-сайт «StoreMobile», який задовольняє потребам замовника.

Методами розробки є використання CMS WordPress та плагіну «WooCommerce» для вирішення поставлених завдань, також будуть використані мови програмування такі як PHP, JS та підключення API від “нова пошта».

Бакалаврська робота складається з таких розділів:

- Вступ
- Інформаційна частина
- Теоретична частина
- Практична частина
- Висновки

Результатом виконанням бакалаврської роботи є створений та розміщений в мережі сайт «StoreMobile».

1. ПОСТАНОВКА ЗАДАЧІ

Перед розробником була поставлена задача замовником проекту створити інтернет магазин для продажу мобільних пристроїв та аксесуарів. Перед початком створення веб-сайту розробник отримав технічне завдання та інструменти для його виконання.

Веб-сайт має бути розроблений на базі CMS WordPress на основі платного шаблону теми яку розробнику передав замовник. Головними вимогами до сайту має бути чітка структура, гарний дизайн, оптимізація, взаємодія карток товарів та корзини між собою, підключення API від «Нова Пошта», яка дозволяє отримувати всі актуальні пункти пошти в режимі реального часу та обирати їх. Для розробника також було поставлене завдання від замовника підключити домен сайту та налаштувати веб-хостинг.

Додатковим завданням було створити адміністративну частину сайту, яка б давала змогу розділяти рівень прав доступу для редакторів сайту та адміністраторів, також керувати динамічними даними. Сайт має бути відкритим для пошуку, оптимізованим для мобільних пристроїв та сучасних інтернет-браузерів, та використовувати фірмові кольори компанії.

Особливості проектування веб-сайту:

- Під час розробки прототипу сайту важливо дотримуватися єдиної стилістики оформлення елементів.
- Для того щоб зробити прототип сайту необхідно розробити його структуру, де чітко позначити ієрархію блоків, розділів і суброзділів.
- Не рекомендується створювати подібні елементи на сторінці.

Наведемо приклад: контакти розміщуються або в розділі про фірму або в майбутні сторінки, тому виділяти для них сторінку просто не має сенсу.

- Якщо функціонал одного блоку дублює функціонал іншого краще видалити його, оскільки повтори заплутують клієнтів які переглядають сайт.

- При створенні прототипу сайту важливо позначити шлях користувача до найцікавіших розділів сайту. Створення прототипу сайту передбачає створення схеми основних сторінок з їх пріоритетністю. Наприклад, якщо фірма не планує часто викладати новини, то немає сенсу верстати для них окрему сторінку.

Крім того, сайт має бути створений згідно технічного завдання, та містити наступні сторінки: «Главная», «О нас», «Контакты».

На головній сторінці мають бути блоки записів, слайдер та останні розміщені товари кожен товар при натисканні має відкривати нову сторінку з описом товару та давати змогу додати товар до кошика.

На сторінці «Про нас» має міститись інформація про компанію яка надаватиме послуги та їх статистику.

На сторінці «Контакты» - має бути інформація про електронну адресу для зв'язку та форма зворотнього зв'язку.

Сайт має коректно відображатись в сучасних веб-браузерах та використовувати кольори та логотипи компанії.

2. ІНФОРМАЦІЙНА ЧАСТИНА

1.1 Поняття веб-сайта

У 21 столітті «технологій» дуже важко уявити мережу інтернет в якому не було б жодного створеного сайту. Розвиток Інтернету відбувається дуже швидкими темпами. Щодня з'являються тисячі нових веб-сайтів, на які може зайти будь-яка людина в цілому світі, при умові, що вона має вихід в Інтернет. На разі створення сайтів можна умовно поділити на декілька видів:

- сайти створені не професіоналами – наприклад блоги домогосподарок або садівників, головна мета яких поділитися власним досвідом, думками, без матеріальної вигоди. Для створення таких сайтів, в наш час прийшли на допомогу онлайн-конструктори сайтів, завдяки яким люди, які навіть не чули слово HTML або PHP всього в декілька кліків миші можуть створити веб-сторінки. Звичайно ми розуміємо, що такі сайти не будуть набувати значної популярності у пошукових алгоритмах, в наш час вони дуже акцентовані на сео-оптимізацію, без яких в наш час сайти будуть просто проігноровані.

Іноді бувають випадки, коли якийсь невеликий блог, створений онлайн-конструктором, набуває популярності завдяки якісному наповненню контентом. Одним з яскравих прикладів створення веб-сайтів створених на онлайн-конструкторі, є створення невеликих інтернет-магазинів, або «landing page». Такі сайти ніколи не зможуть конкурувати з професійно створеними інтернет-магазинами, бо вся проблема в тому, що ці сайти не мають відкритого коду та містять в собі безліч не потрібних «важких» елементів, які відповідають стандартам пошукових алгоритмів [9].

Отже, сайти створені онлайн конструкторами, такими як – «Wix», та інші аналогічні, є лише невеликим інструментом для власників цих сайтів, де вони діляться своїми думками та інформацією, але не більше.

- сайти, створені студентами та впевненими користувачами ПК. При створенні таких сайтів розробники-початківці вже ознайомлені з такими основними технологіями: HTML, PHP, CSS, JS. На створення таких сайтів потрібно на багато більше часу, навичок та вмінь. Створення таких сайтів являє собою написання HTML коду-головної розмітки сайту, разом з використанням стилів CSS. Таким чином, ми можемо отримати простий сайт, який буде відповідати пошуковим алгоритмам та може бути більш конкурентно спроможним, але це тільки візуальна частина сайту, яка не дає змоги користувачу прикріплювати документи, реєструватися на веб-ресурсі, або виконувати якісь інші операції.

Для наповнення сайту функціоналом розробники-початківці можуть створювати власний PHP -код та JS скрипти, які можуть давати можливість сайту бути функціональним, наприклад: розрахувати вартість товару, вибрати пункти відправки, та багато іншого. Якщо додати на сайт JS скрипт код, то замість простої візуалізації сайту ми можемо отримати досить гарний інтернет-ресурс (додати «слайдер», «впливаючі вікна», плавні анімації, прокрутки) [2].

Отже, створення таких сайтів потребує значного часу та зусиль розробників-початківців, але ми маємо відкритий код та можливість в будь-який час змінювати сайт, і його наповнення, і краще підпадати під суворі пошукові алгоритми.

- сайти створені професіональними розробниками та ІТ-компаніями. Підхід до створення сайтів значно відрізняється від попередніх типів розробників, завдяки використанню інженерного підходу та використанню самих сучасних веб-технологій таких, як:

- використання фрейм-ворків;
- написання власних БД;
- нормалізація БД;
- використання сучасних бібліотек, для PHP та Java Script;
- розробка унікального дизайну;
- верстка сайту;
- розробка логотипу;
- використання захисту інформації;

- використання пре-процесорів;
- сео-оптимізація;
- професійне QA-тестування;
- використання СММ та інтернет-маркетингу.

Отже, веб-сайт – сукупність веб-сторінок, доступних в Інтернеті, які об'єднані як за змістом, так і навігаційно. Web-сайт стає в пригоді як інструмент у рішенні освітніх, маркетингових, іміджевих проблем, комунікаційних та будь-яких інших проблем.

Сайт, як і будь яка інформаційна система, має власні цикли розвитку: розроблення концепції, проектування, дизайнерське оформлення, структурування, наповнення контенту, впровадження, редизайн або модернізація в подальшому, що потребує систематичної роботи на кожному етапі. Вони можуть мати різне спрямування, структуру, типи, класифікацію, призначення, наповнення, і їх якість та актуальність залежить від професіональності, та креативності розробника.

1.2 Основні етапи створення веб-сайтів

Гарантією створення якісного сайту може стати правильна і послідовна робота по розробці сайту плюс відточена до дрібниць робота на кожній стадії його створення. Хтось думає, що створювати веб-сайти зовсім не складно, але для створення дісно якісного сайту, необхідно скласти план та розбити його на такі етапи:

- Постановка цілей і розбір завдань, які необхідно виконати, щоб досягти головного результату;
- Вибір типу сайту після визначення цілей. Після складання портрета можна приступати до формування структури сайту, опрацювання оформлення і функціонального навантаження, підборі контенту;
- Опрацювання технічного завдання є наступним етапом. Саме в ТЗ прописуються всі необхідні характеристики сайту: вимоги і побажання по дизайну

сайту, семантика; будова сайту, число сторінок, блоків і категорій; функціональне наповнення сайту; використовувані технології; технічні вимоги до сайту.

- Зразок дизайну. Жоден етап в розробці сайту можна пропустити, але створення оформлення сайту є одним з найважливіших етапів. Перше, на що звертає увагу відвідувач, потрапивши на сайт, – це його оформлення. Саме керуючись своїм першим враженням, користувач вирішує залишатися на сайті або покинути його.

- Верстка. Іншими словами це процес можна назвати як злиття дизайну з двигуном. Верстальник, використовуючи HTML, створює з уже готового дизайну робочий проект сайту [17].

Сайт верстається таким чином, щоб він міг однаково добре відображатися в будь-яких інтернет-браузерах. Стили CSS також створюються на цьому етапі. Наступним кроком на цьому етапі є підключення сайту до CMS. Тепер можна приступати до створення сторінок, категорій і підкатегорій, розставляти гіперпосилання і визначати, як буде функціонувати вкладка меню.

Категорії, підкатегорії, контент, що розміщується на сторінці, товарні картки і т.д – все це верстається на даному етапі створення сайту. Кожен з елементів важливий і має свої вимоги.

Якщо замовник хоче охопити більшу кількість потенційних клієнтів, то повинен розуміти, що розробка адаптивної версії сайту необхідна. Величезна кількість користувачів заходить на сайт з мобільних пристроїв. Адаптивна версія сайту не є обов'язковою, але при бажанні клієнта вона створюється і є останнім кроком на даному етапі.

- Програмування сайту. Програміст бере участь в проекті тільки в тому випадку, коли йде розробка повноцінного великого сайту. В іншому випадку послуги цього фахівця не потрібні, оскільки він розробляє складні функціональні блоки, наприклад, валютний конвертер, калькулятор, системи оплати і так далі. Іншими словами, програміст розробляє додатковий функціонал для сайту, який не є обов'язковим і не входить в звичайний пакет.

- Розміщення контенту Це, по суті, останній крок в роботі над сайтом. На ресурсі розміщується весь необхідний матеріал (фотографії, відео, тести, описи, статті), який буде цікавий і необхідний відвідувачеві.

Забігаючи наперед і передбачаючи майбутнє просування ресурсу, на сайті викладаються оптимізовані з точки зору seo тексти і статті, в яких будуть грамотно прописані потрібні ключі з семантичного ядра. Менеджер, що займається контентом, розміщує різні графічні елементи на потрібних сторінках, після чого всі сторінки сайту пройдуть внутрішню оптимізацію.

- Тестування сайту. Без тестування сайту неможливо визначити, чи коректно працює сайт, чи не допущено будь-яких помилок при його створенні. Тестувальник ретельно перевіряє всю працездатність сайту, керуючись певними критеріями. Якщо помилки в процесі тестування будуть виявлені, то вони усуваються фахівцями, які працюють над створенням сайту [8].

Ще один важливий момент, який можна зробити в будь-який момент. Йдеться про вибір доменного імені та хостингу для сайту. Коли робота над створенням сайту буде закінчена, клієнтові надаються всі логіни і паролі для CMS. Отримавши готовий сайт в користування, замовник може продовжити співпрацю з агентством, замовивши в ньому супровід сайту або його просування.

- Додаткові роботи. Вкрай рідко замовник припиняє співпрацю з агентством після отримання сайту і доступу до нього на руки. Зазвичай вони замовляють в агентстві ще деякі послуги, без яких не можуть обійтися.

До таких відносяться:

- Адміністративна підтримка сайту.
- Написання seo-оптимізованих текстів, статей, описів і т.д.
- Оновлення матеріалів на сайті, в тому числі заповнення каталогів і товарних карток.
- Розкрутка ресурсу.

Замовник може контролювати всі етапи створення сайту. Після закінчення одного етапу і перед початком іншого клієнту надається звіт про виконану роботу, і демонструються результати роботи. Без затвердження клієнтом результатів

попереднього етапу команда фахівців не береться до наступного. Така співпраця дозволяє проконтролювати замовнику роботу агентства, а фахівцям зрозуміти, чи задоволений замовник їх роботою.

1.3. Обґрунтування потреби в веб-сайті

На сьогоднішній день, під час стрімкого розвитку технологій та ринку послуг, важко уявити існування будь-якого виду бізнесу без веб-інтеграції та діджеталізації. Розвиток інтернет технологій та інтернет-маркетингу дає змогу реалізувати будь-який вид товару та підвищити попит на надання послуг. Іншими словами, можна сказати, якщо вашого бізнесу немає в мережі – його немає взагалі.

Тому, веб-сайт є сучасним каналом комунікації з майбутнім споживачем. Адже створення сайту один з найпростіших варіантів заволодіння увагою користувача в інтернеті. В наш час важко знайти користувачів що не шукають потрібну інформацію чи товар Інтернеті. Використовуючи веб-сайт, користувач може замовити потрібний йому товар не виходячи з дому, що дозволяє продавцю зекономити на часі та приміщенні для магазину.

Задля привернення уваги до свого продукту й розширення цільової аудиторії, власник сайту в Інтернеті вдається до все новіших форм реклами та методів просування, які залежать від цільової аудиторії та загальної концепції веб-сайту [4].

Скажімо, промо-сайти – різновиди електронної сторінки, повністю присвяченої рекламній акції із оригінальним дизайном, цікавим функціоналом та текстом, обмеженої в часі й призначеної лише для ознайомлення, суттєво відрізняються від корпоративного веб-сайту, рекламні засоби якого обмежені лише товарами та послугами власне компанії-виробника й закриті для реклами «ззовні». Або ж від простого шаблонного сайту-візитки із усіма можливими видами реклами, поява якої закриває текстові блоки та знижує ефективність читання.

Вибір самого типу реклами залежить, насамперед, не стільки від мети повідомлення, адже мета реклами – спонукати до використання товарами чи послугами, скільки від цільової аудиторії та місця її перебування в Інтернеті.

Вдале, продумане застосування реклами в Інтернеті завжди підвищує довіру потенційного чи існуючого споживача продукту до неї.

Адже, не зважаючи на те, який тип реклами застосовується – брендинг-реклама, яка не очікує від користувача миттєвої реакції, а розраховує на довгостроковий вплив, чи реклама продажів, яка чекає від користувача конкретних миттєвих дій – натиснути на банер, відправити замовлення, подзвонити за номером тощо. Товар чи послуга, якщо реклама не є вдалою, не виконує своїх функцій, або занадто нав'язлива, не буде придбаний, або залишиться поза увагою споживача.

Ще на етапі розроблення закладається майбутнє ефективне функціонування сайтів, оскільки вдало розроблений дизайн, професійно виконана верстка та програмування забезпечують користувачам умови максимальної зручності. Оскільки інтернет-бізнес розвивається надзвичайно швидко, то й кількість комерційних сайтів збільшується стрімко, що зумовлює потребу у формуванні універсальної системи кількісної оцінки якості інтернет-ресурсу [5].

Отже, створення якісного веб-сайту не достатньо для остаточного залучення великої кількості покупців та відвідувачів. Необхідно створити та як можна більше поширити якісну, інформативну та привабливу рекламу, що зацікавить людей та приверне увагу до товару.

Який саме вид інтернет реклами буде доцільним у використанні та залучить багато клієнтів повинен вирішувати власник сайту, бо це залежить від виду товару та цільової аудиторії на яку налаштований вид продукції, або послуг.

2.4. Етапи розробки веб-сайта

2.4.1. Вибір програмних засобів для розробки веб-сайта

Якість і повнота результатів завантаження залежить від програм, що використовуються та постійно вдосконалюються. Результат роботи веб-краулера – статичне представлення веб-сайту, незалежно від того, яким він був до копіювання.

Статичними називаються веб-сайти, інформація в яких зафіксована у формі HTML-сторінок, без використання додаткових програмних засобів для їх зміни за запитами користувачів. При кожному звертанні до веб-сайту для користувачів відтворюються відповідні HTML-сторінки.

Щоб оновити інформацію на подібних сторінках, необхідно вручну внести зміни безпосередньо в HTML-код сторінки. На відміну від статичних, динамічні веб-сайти більш гнучкі в керуванні. Для роботи динамічних веб-сайтів використовуються різні технології, що дозволяють “конструювати” веб-сторінки “на льоту” на запити їх користувачів, а потім відтворювати їх. Динамічні веб-сайти можуть “підлаштовуватися” під своїх відвідувачів, реагуючи на їхні дії. Для цього використовуються технології серверних, клієнтських скриптів, за допомогою яких і створюються сценарії роботи веб-сайтів при певних діях користувачів [16].

Наприклад, сторінка у соціальній мережі, де відображено інформацію, пов’язану з користувачем, який увів свої ідентифікаційні дані.

Пропонуємо визначити такі критерії вибору програмних засобів для копіювання веб-сайтів:

- повнота інформації, що завантажуюється;
- багаторівневість копіювання інформації;
- можливість довантаження інформації з місця зупинки завантаження;
- можливість налаштування параметрів завантаження;
- відкритість програмного коду;
- безпека.

Орієнтація на ці критерії – дуже важлива умова під час вибору програмного засобу копіювання веб-сайтів, адже саме від нього залежить ефективність створення архівних колекцій веб-сайтів.

Розглянемо визначені критерії більш детально:

Повнота інформації, що завантажується. Критерій, викликаний необхідністю завантаження всієї інформації із вказаного веб-сайту, тобто всіх файлів, що знаходяться на ньому, незалежно від його складності. Разом з тим, часто файли зображень, відео, аудіо закриті для завантаження власниками веб-сайтів з метою захисту контенту. Такі файли доводиться завантажувати вручну.

Багаторівневість копіювання інформації. Дуже важливо, щоб у програмі копіювання веб-сайтів була можливість вибору рівнів завантаження, тобто в архівіста була можливість контролю меж переходу програми за посиланнями як всередині одного домену, так і на зовнішні ресурси відносно до нього. Оскільки для створення тематичних колекцій веб-сайтів необхідно копіювати лише інформацію, що тематично належить до створюваних архівних колекцій [7].

Можливість довантаження з місця зупинки завантаження інформації. Утрати Інтернет-з'єднання з будь-яких причин під час завантаження веб-сайту можуть призвести до припинення копіювання веб-сайту.

Тому необхідно, щоб веб-краулер мав змогу продовжити копіювання веб-сайту з того місця, на якому сталася зупинка завантаження інформації.

Можливість налаштування параметрів завантаження. Вебкраулер повинен забезпечувати вибір таких параметрів завантаження веб-сайту: кількість одночасних з'єднань та кількість повторів автоматичного з'єднання, якщо відбулася втрата Інтернет-з'єднання.

Необхідність наявності такого параметру, як кількість одночасних з'єднань, зумовлена тим, що деякі веб-сайти мають захист від перенавантаження каналу. Для цього встановлюється ліміт на загальну кількість одночасних звернень користувачів до контенту веб-сайту. При перевищенні цього ліміту, всі запити від джерела, що викликало перенавантаження, на практиці блокуються.

Тобто, у випадку з веб-краулером, копіювання веб-сайту припиняється на період блокування. Це збільшує час, необхідний для повного копіювання веб-сайту. Відкритість програмного коду.

Важливість відкритого програмного коду полягає у незалежності від розробника програмного засобу, що застосовується для копіювання веб-сайтів. Удосконалення програми із закритим кодом та внесення змін може здійснювати лише розробник. У випадку завершення підтримки програмного продукту, з часом він стає застарілим та неактуальним для використання.

На відміну від програм із закритим кодом, програми з відкритим кодом можуть бути модифіковані або допрацьовані, вільно розповсюджуватися, адаптовані для власних потреб. Завдяки цьому, такі програми частіше оновлюються, їх помилки швидше виявляються та виправляються [11].

Блокування зовнішніх посилань. Оскільки на веб-сайті можуть бути розміщені посилання на веб-ресурси, що не є необхідними для завантаження; виходячи з тематики створюваної колекції веб-сайтів, потрібно, щоб веб-краулер блокував можливість переходу за такими посиланнями у створених копіях.

Важливим чинником є те, що зовнішні посилання повинні бути ізольовані, а не видалені взагалі. Тобто при переході за ними виводилось повідомлення про те, що ці посилання ведуть до веб-ресурсів, які не є частиною скопійованого веб-сайту.

Безпека. Необхідно, щоб програмний засіб був перевірений на безпечність і виконував лише ті функції, які повинен виконувати. Досягти цього можна створенням комплексної системи захисту інформації в установі в порядку, визначеному законодавством України.

2.4.2. Обґрунтування вибору CMS

Перші системи управління контентом з'явилися на початку 2000 року. У той час вони були дуже вузьконаправленими рішеннями і в основному вирішували завдання управління одним сайтом. Трохи пізніше CMS стали створюватися в більш універсальному вигляді, тобто з можливістю підстроювання під конкретний проект.

Еволюція CMS відбувається досить швидко, і це в першу чергу пов'язано з появою нових течій і стандартів, наприклад, поява методики AJAX на поточний момент дає можливість уявити інтерфейс CMS практично не відрізнитись по інтерактивності від звичайних desktop-додатку [3].

CMS – це система управління контентом сайтів, покликана максимально спростити і автоматизувати процес редагування окремих web-сторінок або всіх сторінок в цілому, а так само окремих категорій сторінок, які бачить на екрані користувач.

Будь-яка система управління контентом сайтів (CMS) виконує безліч функцій, необхідних для управління сайтами різних видів:

- дозволяє створювати, видаляти і впорядковувати сторінки сайту;
- формує зручні засоби навігації по сайту (посилання, меню, карту сайту, списки сторінок, покажчики пройденого шляху);
- дозволяє створити розділи різних типів (статті, новинні стрічки, форуми, дошки оголошень, поштові форми, опитування, голосування тощо);
- наповнює сторінки сайту блоками контенту різних типів (текст, зображення, список, таблиця);
- забезпечує колективне управління вмістом сайту за рахунок поділу доступу адміністраторів до різних частинами сайту (розділах і групах розділів).

В наш час дуже часто перед розробником виникає питання які інструменти розробки обрати, аби вирішити те чи інше питання, в наш час з'явилося дуже багато систем контролю вмістом (CMS), в яких є як сильні сторони так і слабкі.

Насамперед треба визначитись якого плану буде сайт, до якого типу він належить, та які цілі стоять у власника сайту. Чи то буде сайт візитка, чи крупний інтернет магазин - все це дуже впливає на вибір CMS.

Дуже часто в мережі інтернет можна зустріти сайти візитки які створені без застосування CMS, лише за допомогою HTML, CSS, JS, PHP, причиною не залучення систем CMS до створення сайтів візиток, є те, що сайти візитки або

"лендінг" сторінки не містять в собі ніякого функціоналу, економлячи простір на сервері та спрощуючи пошук в мережі.

Сайти для яких найкраще може підійти CMS:

Для інтернет магазину дуже гарним варіантом CMS буде Joomla або Drupal, ці CMS були спеціально створені під великий бізнес та мають гарну оптимізацію при роботі з великими об'ємами даних мінусом є те, що працювати з цими системами можуть розробники більш високого рівня, адже системи не мають легкого інтерфейсу.

Для блогу новин або сайту портфоліо і маленьких інтернет магазинів буде достатньо WordPress, система має легкий інтерфейс та не важка в вивченні, ця CMS відкриває безліч можливостей та готових рішень, але мінусом є те, що будь яку сторінку на WP потрібно оптимізувати як в плані SEO так і функціонально [1].

В Сполучених Штатах та Європі дуже популярним CMS є Magento та PrestaShop – ці системи мають дуже великий функціонал та достатньо важкі в вивченні, але вони дають змогу вирішувати проблеми великого бізнесу та полегшувати роботу з сайтом компанії.

Для сайтів які потребують обліку та аудиту і праці з великими базами даних, та не потребують глибокого розуміння системи ідеально підійде CMS Bitrix 24, яка дуже популярна в країнах СНГ.

Розуміючи потребу нашого замовника в розробці невеликого інтернет магазину, та на підставі технічного завдання було обрано CMS WordPress, яка допоможе реалізувати необхідний функціонал, та спростить роботу майбутнім адміністраторам магазину.

Більш детально розглянемо питання актуальності WP, чому саме ця система одна з найпопулярніших в світі, справа в тому, що WP безкоштовна, та дає змогу використовувати величезну безкоштовну базу безкоштовних шаблонів тем, або власноруч створити шаблони.

WP система яка написана мовою PHP та має відкритий код, тобто при необхідності ми можемо залізити в "ядро" та налаштувати його так, як нам потрібно,

також використовуючи цю CMS, ми можемо бути впевнені в захисті від спам-ботів та постійних оновленнях захисту.

Ще одним важливим фактором є те, що WP має можливість підключати плагіни, які дозволяють розширювати функціонал вашого сайту навіть не знаючи мови програмування PHP [14].

Отже, запорукою успіху при створенні веб-сторінок є чітко створений проект майбутнього сайту, який вміщує в собі всі можливі побажання замовника при можливостях розробника, а також методологія розробки сайту, чи це буде сайт створений власноруч за допомогою HTML, CSS, PHP, JS чи це буде сайт на базі CMS яка в свій час може виконати поставлені задачі згідно технічного завдання.

Використовуючи сучасні технології в розробці веб-сайтів можна в багато разів прискорити як розробку самого веб-ресурсу, так і його оптимізацію, що в свій час дасть змогу підвищити продажі компанії або збільшити кількість користувачів.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1 Розробка веб-дизайну сайту

Часто перше знайомство з компанією для потенційного клієнта відбувається саме через сайт. При візуальному ознайомленні користувач вже за перші 2-3 секунди формує для себе враження про фірму. Це враження цілком може стати ключовим при виборі компанії для співпраці. Буде зроблений вибір на Вашу користь чи на користь одного з Ваших конкурентів – залежить безпосередньо від якості сайту.

Веб-дизайн визначає зовнішній вигляд сайту, це перше, що бачить відвідувач. Таким чином, дизайн сайту вже підсвідомо формує думку про підприємство у потенційних клієнтів.

Дратівливими або переобтяженими деталями веб-дизайн може відштовхнути відвідувача від подальшого вивчення сайту, і, можливо, іншого шансу зацікавити цю людину не буде.

Ми сприймаємо веб-дизайн сайтів як рішення задачі, де умови – це вимоги клієнта, сфера, діяльності, конкурентне середовище, а відповідь – це зручний і красивий дизайн сайту. Саме тому ми з усією відповідальністю ставимось до етапу збору інформації, аналізу конкурентів і цілей замовника. Як бачите, веб-дизайн сайту – це не просто малювання. Будь-який дизайн сайту передбачає роботу в кілька етапів і вимагає зусиль не тільки з боку виконавця, але і зацікавленість з боку клієнта [10].

Якщо у вас вже є своя технологічна платформа для сайту, але немає для нього дизайну – дизайн студія може прийняти замовлення тільки на розробку веб-дизайну.

У результаті, Замовник отримує не просто «красиву картинку», а повністю зверстану HTML-копію майбутньої веб-сторінки з необхідним JavaScript-кодом і Flash-включеннями. У пакет створення веб-дизайну входить консультаційна допомога з інтеграції дизайну в програмне забезпечення стороннього виробництва.

Розробка веб-дизайну (як і розробка сайту) починається з підготовки та затвердження прототипу, що дозволяє зрозуміти структуру і продумати ергономіку майбутньої системи. Початок будь-якої креативної концепції – фундаментальне вивчення всіх складових Вашої рекламної компанії, від вивчення цільової аудиторії і позиціонування, до поточних акцій.

Завдяки новим технологіям ми отримуємо все більш спрощені інструменти для того, щоб реалізувати сайт дизайнерами. Якщо ж подивитися з іншої сторони, то цей процес вимагає знань, вмінь та навичок, які фахівець має почерпнути з різних наук. Так, вмілий дизайнер не лише має вміти гарно малювати та передавати ідеї, а і гарно володіти знаннями комп'ютерних технологій та добре знатись на маркетингу. Фахівець має володіти розвинутим асоційованим мисленням, а також бути інтелектуально обізнаним в різних сферах [12].

Таким чином ми бачимо, що веб-дизайн це не так просто і знайти хороших спеціалістів в наш час доволі складно. За допомогою ви можете звернутися до веб студії.

3.2 Структура сайту

Структура сайту – це внутрішній устрій сайту, розташування сторінок, розділів, підрозділів, додаткових матеріалів. І першочерговим завданням дизайнера є створення стрункого порядку з хаотичного скупчення інформації.

Структура сайту інтернет-магазину впливає на простоту використання ресурсу. Логічна і зручна структура дозволяє користувачам легко знаходити потрібні товари. В кінцевому підсумку це впливає на конверсію, задоволеність відвідувачів і навіть пошуковий трафік. А ці фактори впливають на фінансові показники магазину: виручку і прибуток [15].

На логічному рівні структура сайту є сукупністю сторінок, які поєднані між собою одним дизайном, стилем і посиланнями. Логічне проектування передбачає організацію інформації на сайті, побудову його структури та навігацію по розділах. Тому, розглянемо логічну структуру на прикладі сайту «StoreMobile» (рис. 3.1).

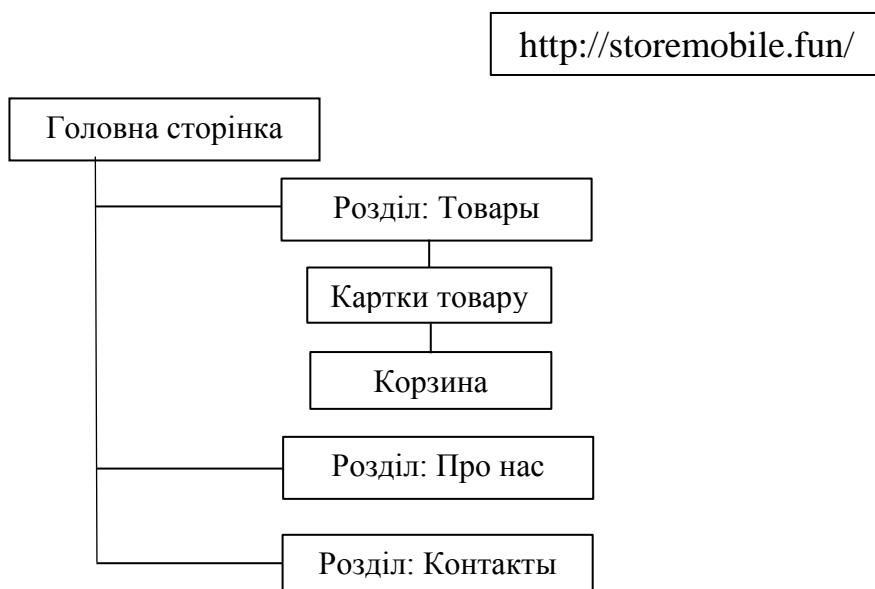


Рис. 3.1 – Логічна структура «StoreMobile»

Фізична структура має на увазі алгоритм розміщення фізичних файлів по піддиректоріях папки, в якій опубліковано ваш сайт.

Очевидно, що логічна і фізична структури можуть не збігатися, оскільки в загальному випадку фізична структура ресурсу розробляється, виходячи зі зручності розміщення файлів. Однак більш-менш точне збереження порядку проходження логічних розділів у фізичній структурі сайту дозволяє уникнути плутанини при подальшому доповненні і оновленні матеріалів.

При створенні фізичної структури слід дотримуватися таких правил:

Потрібно призначати імена директорій, імена і розширення документів HTML і графічних файлів з використанням символів тільки латинського алфавіту і тільки в рядковому регістрі. Намагатися, щоб імена створених вами файлів і директорій не перевищували по довжині восьми символів.

Для того щоб полегшити процес оновлення веб-сторінок, додатки розділів або створення нових рубрик, рекомендується створити засіб документування проекту - будь-яку електронну або звичайну таблицю, в яку записуйте відповідність елементів фізичної структури вашого проекту його логічну структуру. Для прикладу приведемо типову структуру сайту «StoreMobile» (рис.3.2).

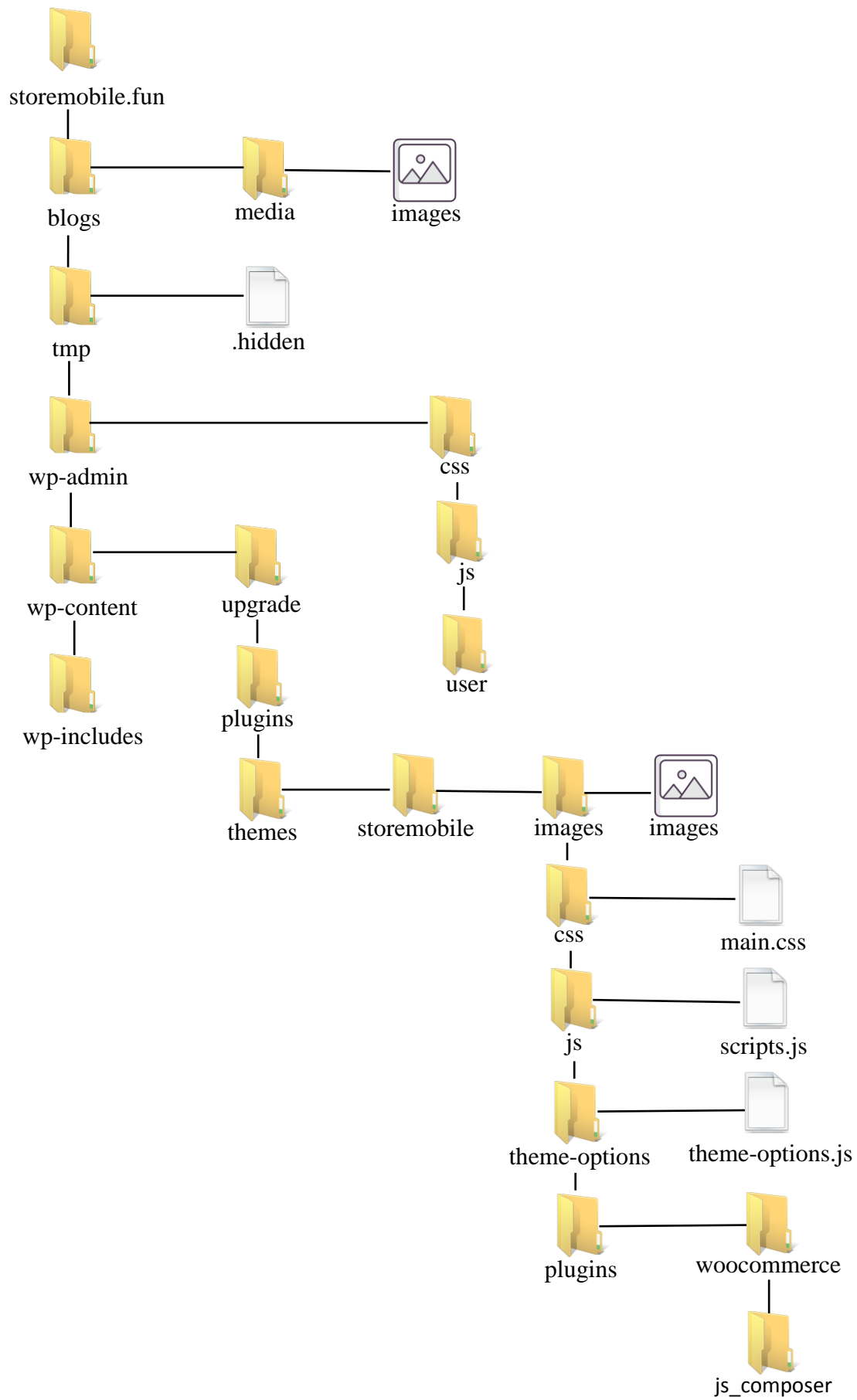


Рис. 3.2 – Фізична структура «StoreMobile»

3.3 Наповнення сайту товарами

Саме контент і являє собою основний інструмент, за допомогою якого можна залучити звичайного (може навіть і випадкового) відвідувача вашого сайту в клієнта, який захоче витратити гроші на придбання пропонованих вами товарів чи послуг.

Самий трудомісткий процес це наповнення інтернет-магазину товаром, де потрібна велика маса контенту, тому дана справа краще довірити контент-менеджеру, який зробить це швидше та якісніше.

Тому якщо вам потрібен повноцінний, працюючий інтернет-ресурс, який буде ефективно працювати пам'ятаєте що розробка сайтів - процедура ресурсномістка, але наповнення сайту не менш важливе. Розробити сайт, який буде забезпечувати отримання серйозної прибутку – це дуже нелегке завдання, але цілком здійсненна.

Звернувшись до професіоналів, ви зможете отримати такий контент, який буде працювати на ваш бізнес, і ніколи не загубиться між багатьма іншими пропозиціями конкурентів. Контент здатний переконати потенційного покупця, що саме ваша пропозиція підходить саме йому. Тому розглянемо контент та його наповнення на прикладі сайту «StoreMobile» (рис 3.3).

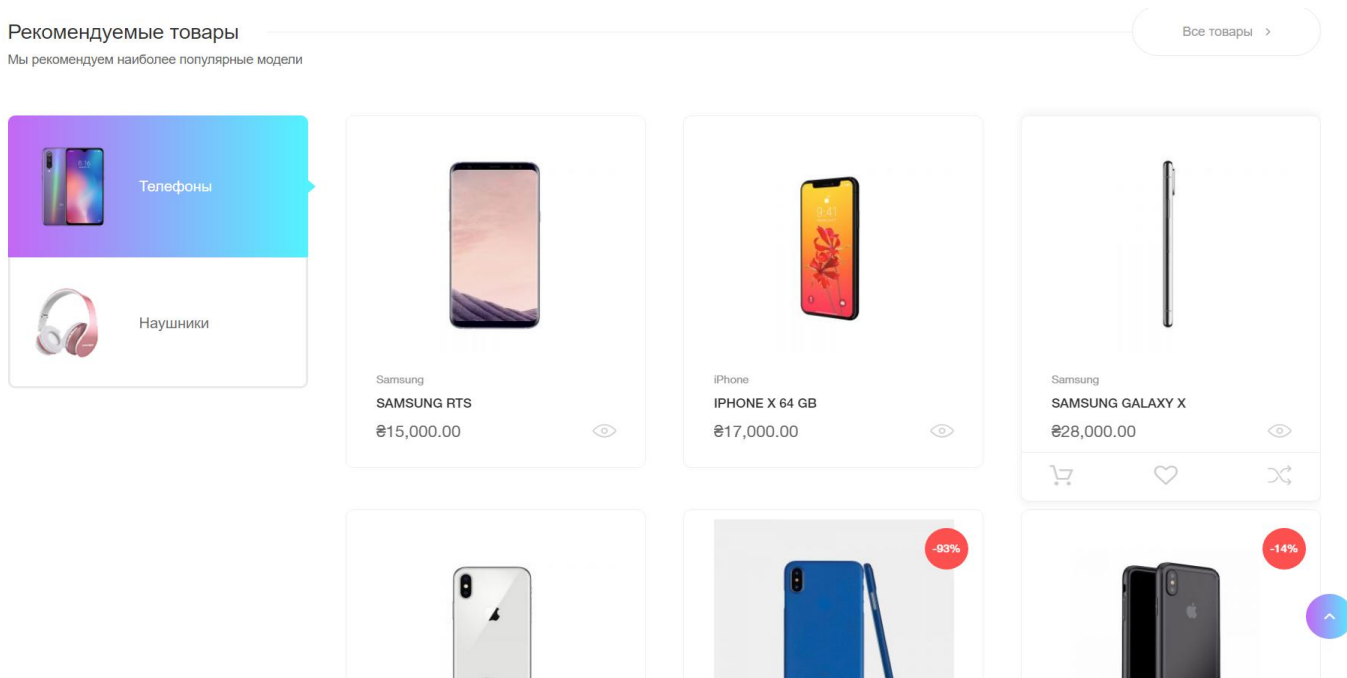


Рис. 3.3 – Наповнення сайту товарами

Інтернет-магазин має надати покупцю більш детальну інформацію про кожен вид товару, в якому необхідно написати технічні характеристики, вартість товару та надати можливість додати товар в корзину, як зображено на рис. 3.4.

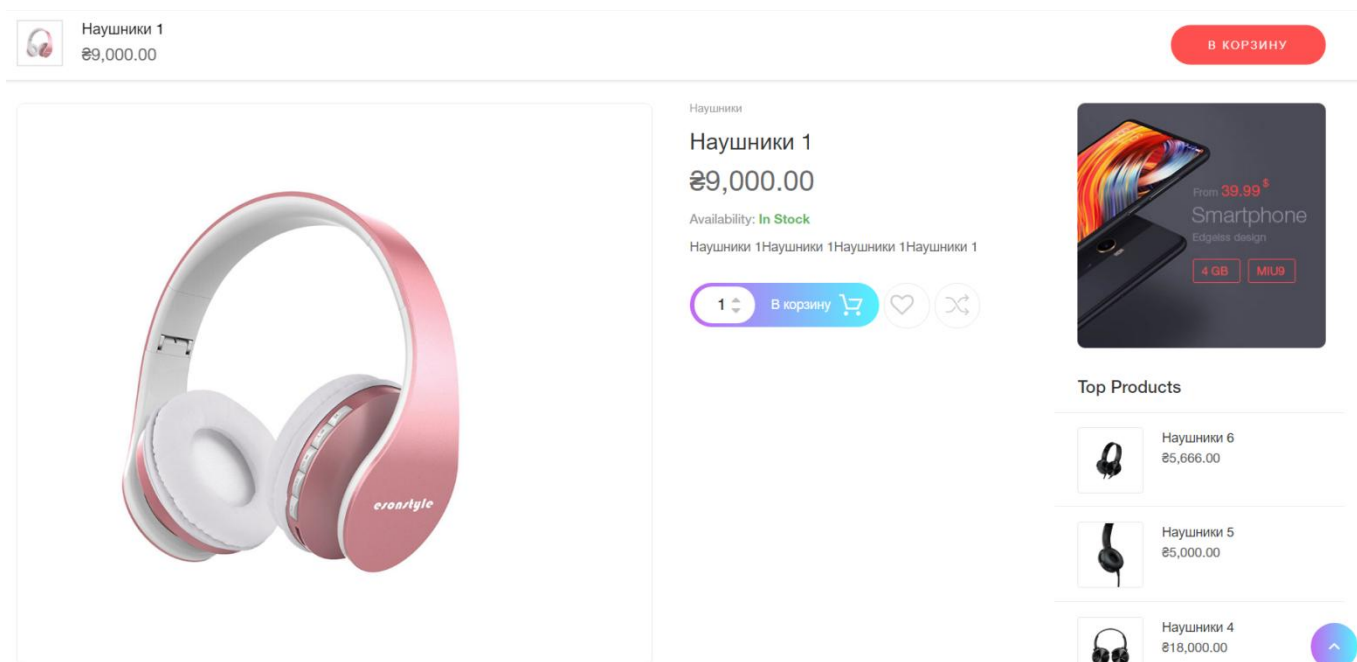


Рис. 3.4 – Сторінка товару

Дуже часто популярні інтернет-магазини розміщують статті з актуальними новинами про тематичні товари, як зображено на рис. 3.5.

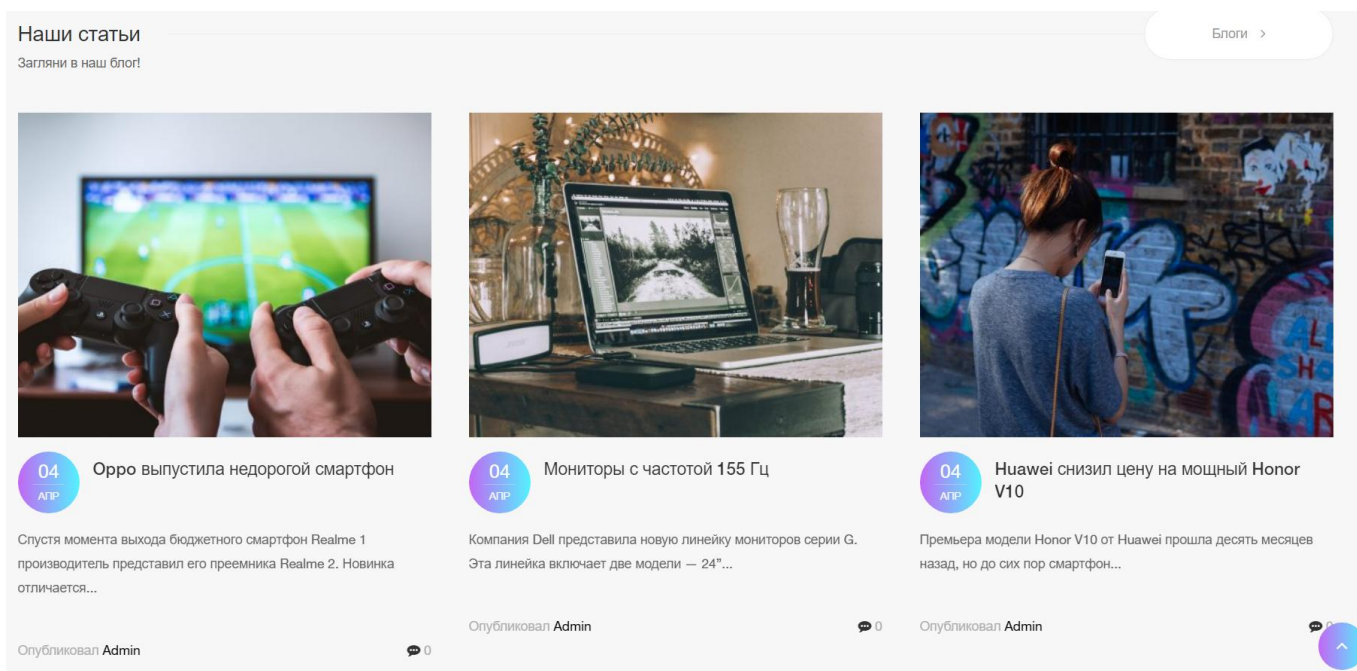


Рис. 3.5 – Блок новин

4. ПРАКТИЧНА ЧАСТИНА

4.1 Встановлення на сервер CMS WordPress

Хостинг надається з панеллю керування – спеціальною програмою, що допомагає управляти хостингом. Розглянемо установку CMS на прикладі ISPmanager, для сайту StoreMobile:

Перший етап – встановлюємо CMS. Натискаємо Скрипти (рис. 4.1).

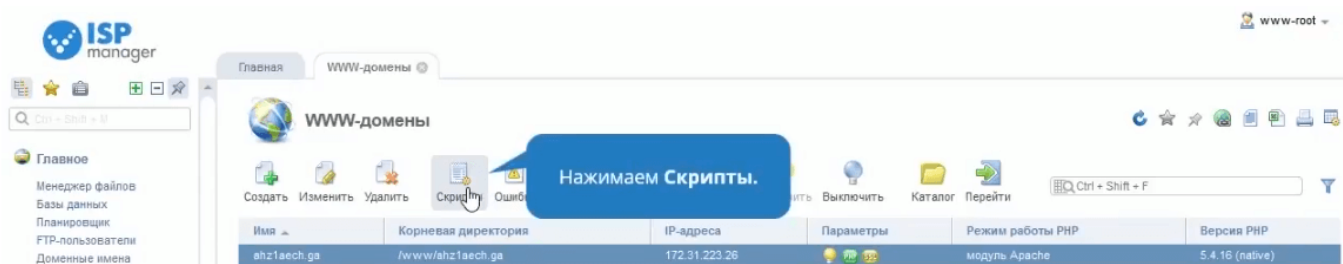


Рис. 4.1 – Початок встановлення CMS WP

Далі налаштовуємо хостинг під CMS: створюємо нову базу даних і користувача бази даних, дані бажано зберегти в окремому файлі, потім стануть в нагоді (рис. 4.2).

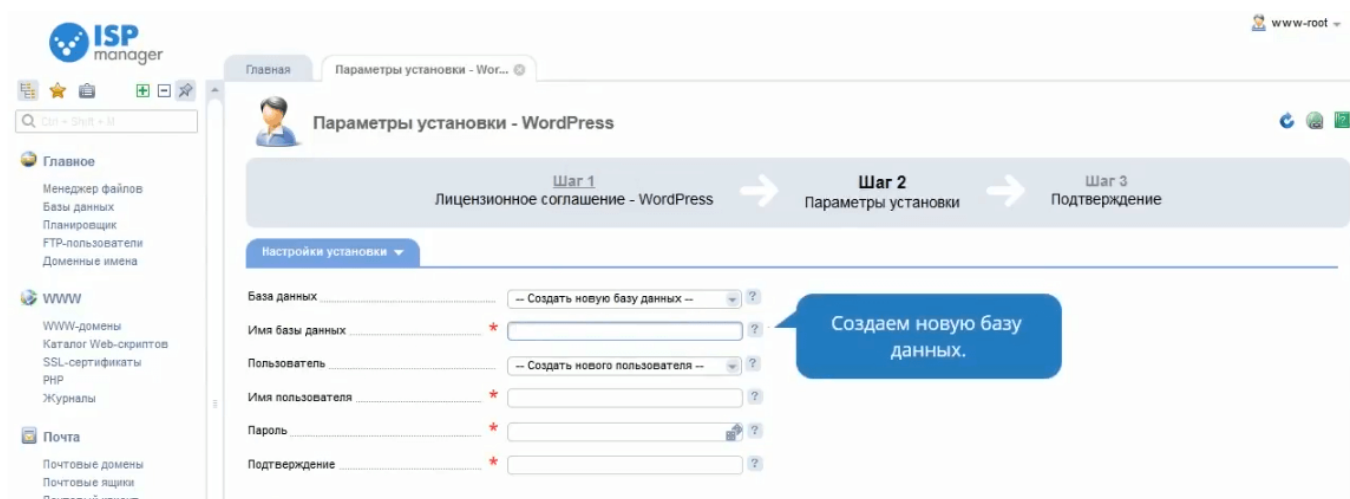


Рис. 4.2 – Налаштування БД

Встановимо пароль для WordPress і збережемо в окремому файлі. Також необхідно вибрати мову інтерфейсу та натиснути «Далі», як зображено на рис. 4.3.

ISP manager

Главная Параметры установки - Wor...

Параметры установки - WordPress

Шаг 1 Лицензионное соглашение - WordPress → Шаг 2 Параметры установки → Шаг 3 Подтверждение

Настройки установки ▼

База данных -- Создать новую базу данных -- ?

Имя базы данных * ?

Пользователь -- Создать нового пользователя -- ?

Имя пользователя * ?

Пароль * ?

Подтверждение * ?

Настройки приложения ▼

Administrator's login admin

Password * ?

Подтверждение * ?

Administrator's email webmaster@vfedoseenko.fvds.ru

Site name My CMS

Interface language English

Рис. 4.3 – Налаштування даних адміністратора

Після всіх налаштувань система повідомить нас про наявність помилок, або про успішне встановлення CMS, як зображено на рис. 4.4.

ISP manager

Главная Подтверждение - WordPress...

Подтверждение - WordPress

Шаг 1 Лицензионное соглашение → Шаг 2 Параметры установки - WordPress → Шаг 3 Подтверждение

Приложение будет автоматически установлено в директорию '/www/ahz1aech.ga'

Если указанная директория не существует, то она будет создана. В противном случае, её содержимое будет перезаписано, что может привести к потере ваших данных

Нажмите кнопку 'Завершить' для окончания установки

<< Назад Завершить Отмена

Почти готово, нажимаем Завершить.

Рис. 4.4 – Завершення установки

4.2 Встановлення платного шаблону сайту

В світі сучасних технологій між розробником та замовником утворилися певні відносини, зараз дуже ціниться економія часу та коштів на розробку продуктів, які зможуть принести фінансову вигоду замовнику. Тобто створення сайту з нуля зараз не є економічно обґрунтованою послугою, натомість створити сайт з платного шаблону, ціна якого коливається від 20 до 100 доларів США. Та налаштування шаблону згідно технічного завдання, буде більш дешевшим та економними.

Саме тому розробнику сайту «StoreMobile» було надано доступ до платного шаблону теми, який необхідно було довести до ладу.

Почати роботу з темою треба з її встановлення та налаштування, встановлювати тему можна декількома способами, наприклад : встановити через адміністративну панель WP, за допомогою FTP файлового менеджера, або на пряму через хостинг сайту.

Розглянемо встановлення шаблону теми за допомогою адміністративної панелі WP в декілька кроків, перший крок це перехід в меню «Внешний вид» та обрати «Теми» як зображено на рис.4.5.

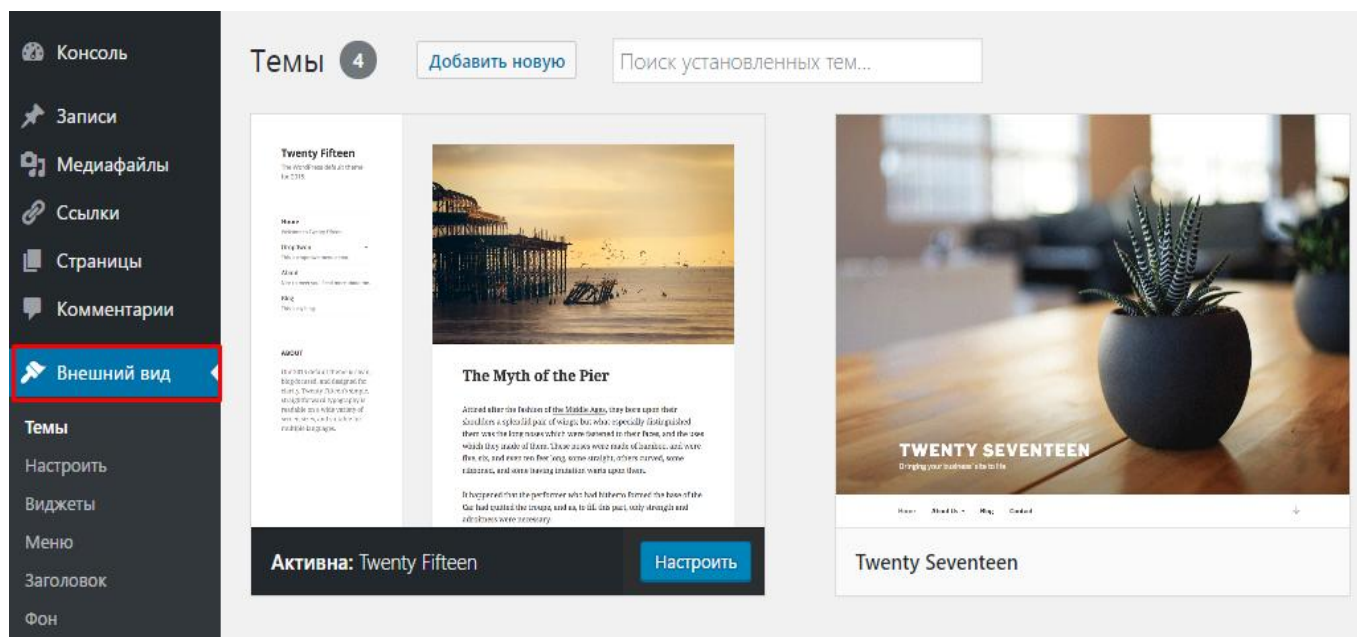


Рис 4.5 – Перший крок встановлення шаблону

Наступним кроком встановлення шаблону, буде загрузка архіва з темою як зображено на рис 4.6.

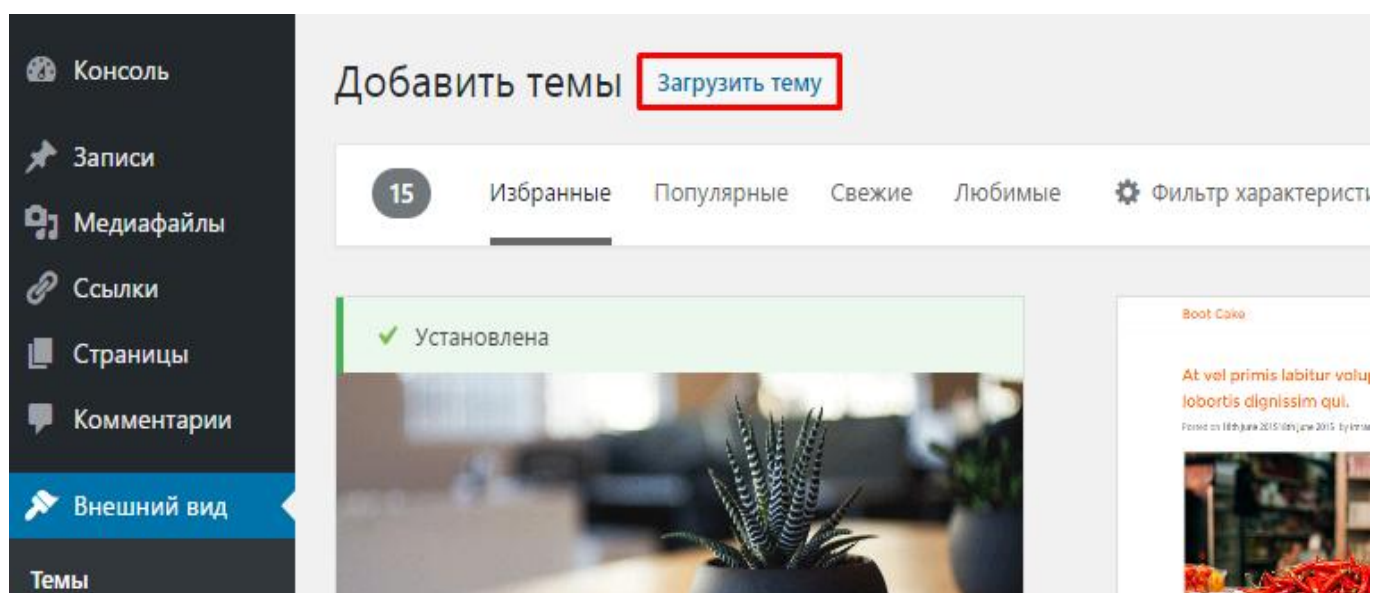


Рис 4.6 – Другий крок загрузка шаблону

Останнім кроком є автоматичне встановлення теми, після якого необхідно буде встановити плагіни та почати налаштування шаблону, як зображено на рис 4.7.

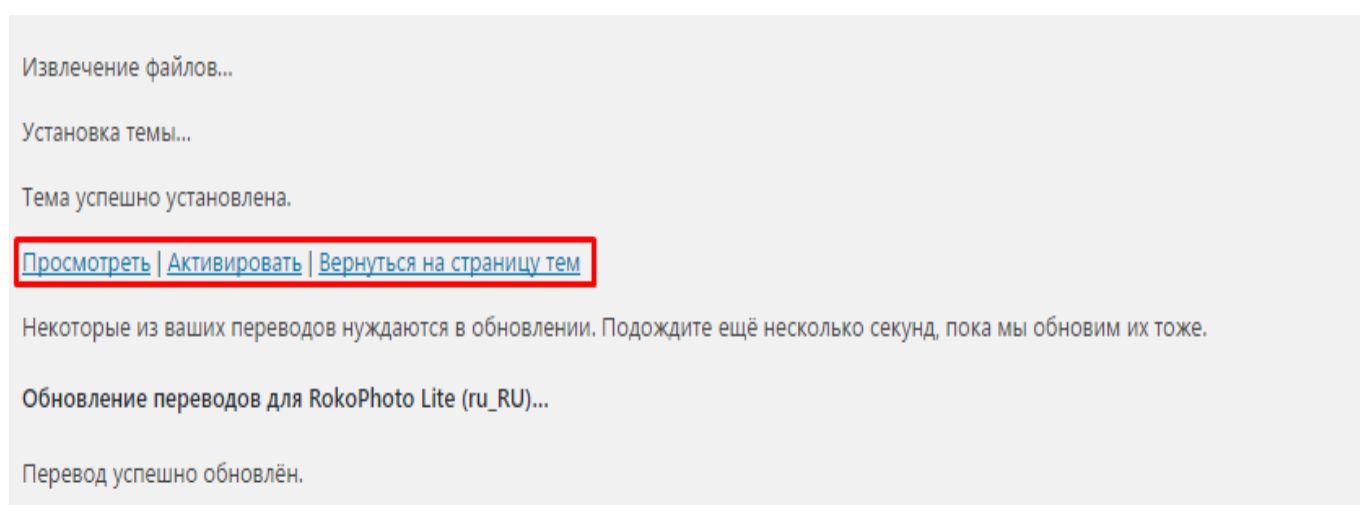


Рис 4.7 – Останній етап загрузки теми

Отже, встановити шаблон за допомогою адміністративної панелі WP дуже легко та швидко, але іноді бувають деякі технічні помилки при встановленні, які має вирішити або розробник, або технічна підтримка хостингу, в нашому випадку ніяких помилок не було, все що необхідно встановити це плагіни та почати налаштування теми сайту.

4.3 Встановлення необхідних плагінів на сайт

В багатьох випадках сучасні сайти мають містити деякий загальний функціонал, який буде спрощувати роботу користувача з сайтом, або ж спрощувати роботу адміністратора сайту, це може бути як автоматичне розміщення постів, сортування розділів, календар подій та багато іншого, всі ці інструменти при розробці самостійно займуть дуже багато часу та потребують високого рівня кваліфікації програмістів. Коли замовник домовляється про створення сайту з розробником на базі CMS, то вже йдеться про використання плагінів, які давали б змогу виконувати всі потреби замовника та відповідати його технічному завданню.

В наш час існує безліч різноманітних плагінів для будь яких цілей, іноді за допомогою плагіна можна створити повністю нову сторінку з інформацією, а іноді плагін може автоматизувати комерційну складову сайту. Розглянемо на прикладі сайту «StoreMobile» роботу плагіну «woocommerce». Встановлення плагіну абсолютно таке ж як і встановлення теми. Зовнішній вигляд плагіну зображено на рис 4.8.

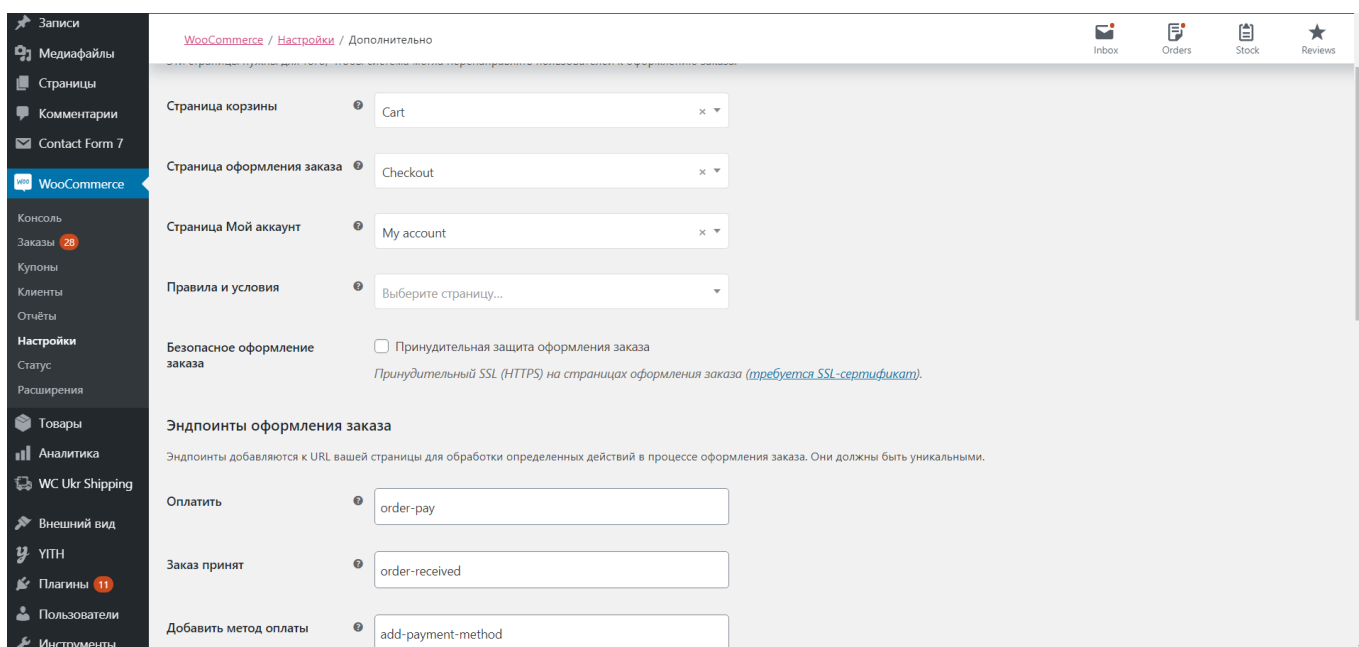


Рис 4.8. – Налаштування плагіну «woocommerce»

Плагін дає змогу одразу автоматично створити як окрему БД для товарів та розмістити її на сайті, так і корзину для цих товарів, для більш точних налаштувань

цього плагіну необхідно налаштування через відкритий код плагіну та файли скриптів, які будуть розглянуті окремо. Продемонструємо роботу плагіну на рис 4.9.

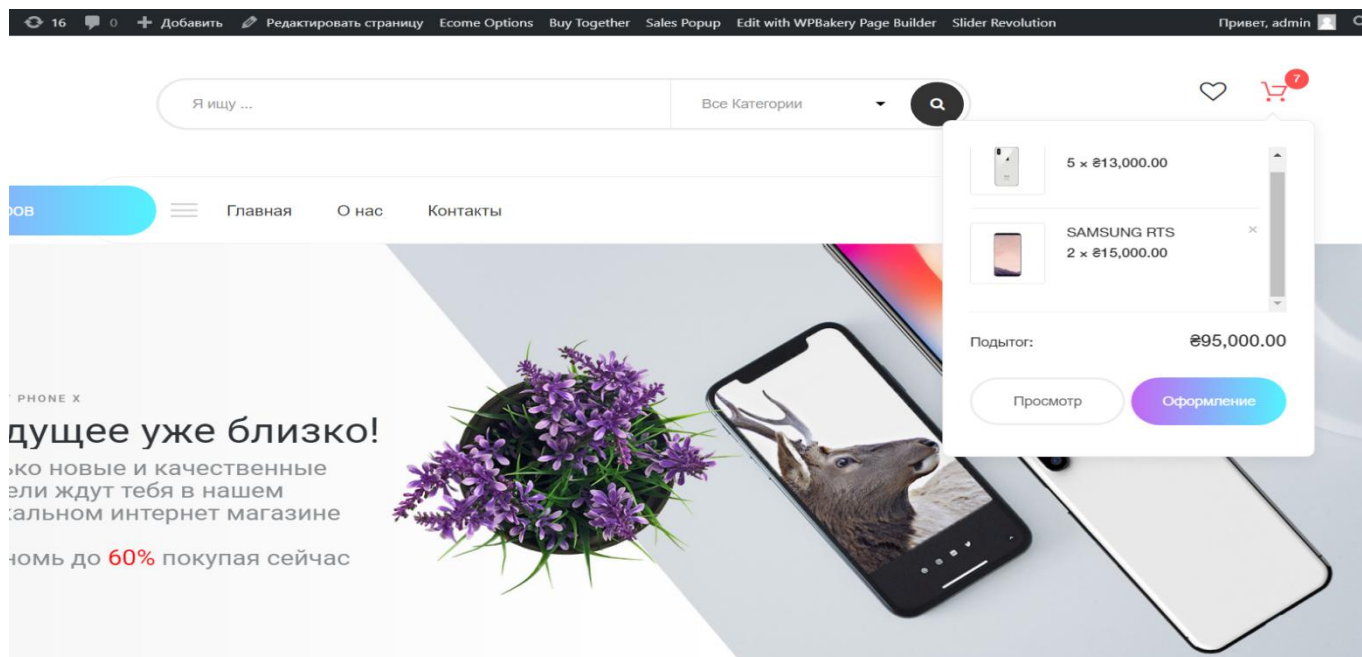


Рис 4.9. – Результат роботи плагіну

Тобто, плагін автоматично створив окрему кнопку для корзини та дав змогу розробнику не витратити час на цей функціонал, розробником було змінено файли стилів цього плагіну для більш коректного відображення на сайті, адже в будь-якого плагіну є свої налаштування стилів, які можуть наслідуватись не вірно, тому без втручання розробника такі помилки виправити не можливо.

4.4. Особливості налаштування теми

Користуючись сайтом на основі шаблону, досить важко працювати з чужим кодом та редагувати його, загалом CMS пропонує нам прямо не виходячи з адміністративної панелі сайту видалити непотрібні сторінки, або якісь зайві матеріали з сайту без втручання в код, знайти потрібний елемент коду може бути досить складно.

Розглянемо можливе видалення непотрібних сторінок, які не підпадають під технічні вимоги до сайту, як зображено на рис 4.10.

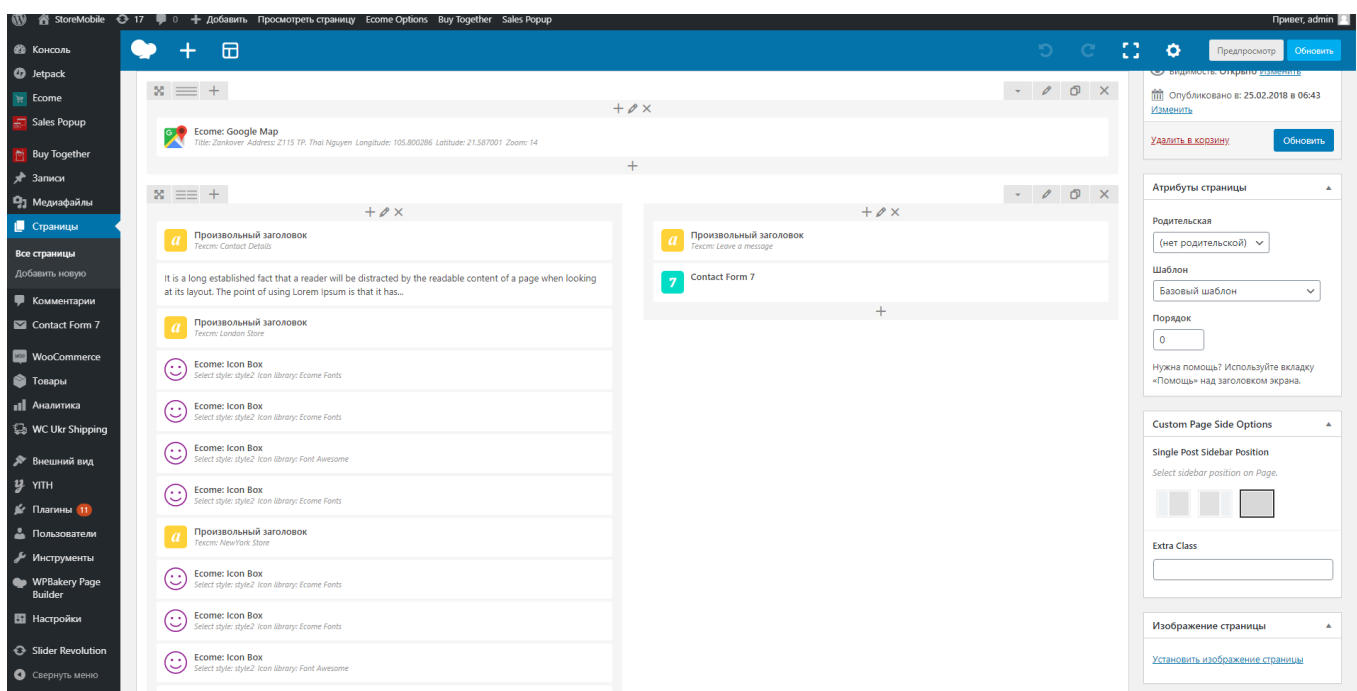


Рис 4.10 – Робота з видалення непотрібних фрагментів на сайті

Звернемо увагу на те, що всі блоки з інформацією побудовані за допомогою плагіну Wp Bakery Page Builder. Роботу з цим плагіном можна віднести до ще одного етапу конструювання сторінок, але зациклювати свою увагу не будемо, краще розібратись поступово з цікавими функціями сайту розглядаючи код елементів.

В хедері сайту звернемо увагу на поле пошуку товарів, яка зображена на рис.4.11.



Рис 4.11 – Поле пошуку товарів

Розглянемо реалізацію з власними правками до коду цієї функції. Для початку перед розробником було завдання русифікувати слова які були в самому полі для вводу, для цього знайдемо через пошук файлів код елементу який відповідає за відображення інформації «плейсхолдер», як зображено на рис 4.12.

```

1 function wpforo_search_form( $html ) {
2
3     $html = str_replace( 'placeholder="Search ', 'placeholder="Я ищу ', $html );
4
5     return $html;
6 }
7 add_filter( 'get_search_form', 'wpforo_search_form' );
8

```

4.12 – Плейсхолдер поля пошуку

Розглянемо саму функцію пошуку, за якою знаходяться товари з БД, та внесемо зміни до функції пошуку, зображено на рис 4.13.

```

1 if ( ! function_exists( 'ecome_search_form' ) ) {
2     function ecome_search_form() {
3         $key_words = Ecome_Functions::ecome_get_option( 'key_word' );
4         $selected = '';
5         if ( isset( $_GET['product_cat'] ) && $_GET['product_cat'] ) {
6             $selected = $_GET['product_cat'];
7         }
8         $args = array(
9             'show_option_none' => esc_html__( 'search', 'ecome' ),
10            'taxonomy'          => 'product_cat',
11            'class'             => 'category-search-option',
12            'hide_empty'        => 1,
13            'orderby'           => 'name',
14            'order'             => 'ASC',
15            'tab_index'         => true,
16            'hierarchical'      => true,
17            'id'                => rand(),
18            'name'              => 'product_cat',
19            'value_field'        => 'slug',
20            'selected'          => $selected,
21            'option_none_value' => '0',
22        );
23        ?>
24        <div class="block-search">
25            <form role="search" method="get" action="<?php echo esc_url( home_url( '/' ) ); ?>"
26                class="form-search block-search ecome-live-search-form">
27                <div class="form-content search-box results-search">
28                    <div class="inner">
29                        <input autocomplete="off" type="text" class="searchfield txt-livesearch input" name="s"
30                            value="<?php echo esc_attr( get_search_query() ); ?>"
31                            placeholder="<?php echo esc_html__( 'Я ищу ...', 'ecome' ); ?>"
32                    </div>
33                </div>
34                <?php if ( class_exists( 'WooCommerce' ) ): ?>
35                    <input type="hidden" name="post_type" value="product"/>
36                    <input type="hidden" name="taxonomy" value="product_cat">
37                    <div class="category">
38                        <?php wp_dropdown_categories( $args ); ?>
39                    </div>
40                <?php else: ?>
41                    <input type="hidden" name="post_type" value="post"/>
42                <?php endif; ?>
43                <button type="submit" class="btn-submit">
44                    <span class="fa fa-search" aria-hidden="true"></span>
45                </button>
46            </form><!-- кінець блока пошука -->
47        </div>
48    }
49}

```

Рис 4.13 – Скрипт пошуку

Більш детально розглянемо цікаві моменти коду :

if (! function_exists('ecome_search_form')) – перевірка на існування функції з таким ім'ям, перевірка створена розробником в цілях безпеки, аби не допускати створення функції з таким ім'ям в подальшому власниками сайту, це звичайна практика в програмуванні.

```
$key_words = Ecome_Functions::ecome_get_option( 'key_word' );
$selected = "";
if ( isset( $_GET['product_cat'] ) && $_GET['product_cat'] ) {
    $selected = $_GET['product_cat'];
```

Створюємо перемінну \$key_words – яка наслідуює від створеної раніше розробниками функцію та записує введену користувачем назву та прирівнює слово до тих, які є в базі товару.

Блок який починається з тегів HTML та ними закінчується розглядати не будемо, це досить не складно зрозуміти прочитавши код.

Залашилось вивести отриману інформацію після пошуку, розглянемо наступний код створений розробником:

```
<?php foreach ( $key_words as $key_word ): ?>
<a class="key-item" href="<?php echo esc_url( $key_word['key_word_link'] );
?>">
<?php echo esc_html( $key_word['key_word_item'] ); ?>
</a>
<?php endforeach; ?>
```

Власне все, що необхідно було зробити для вдалого відображення пошуку – створити цикл «foreach» та порівняти між собою змінну яка зберегла введені дані користувачем, та змінну яка відповідає назві товарів та вивести на екран відповідь пошуку. Якщо до цього коду додати елемента JS, CSS які наслідуються завдяки використанню вірно створеного шаблону, то отримуємо наступний результат, рис 4.14.

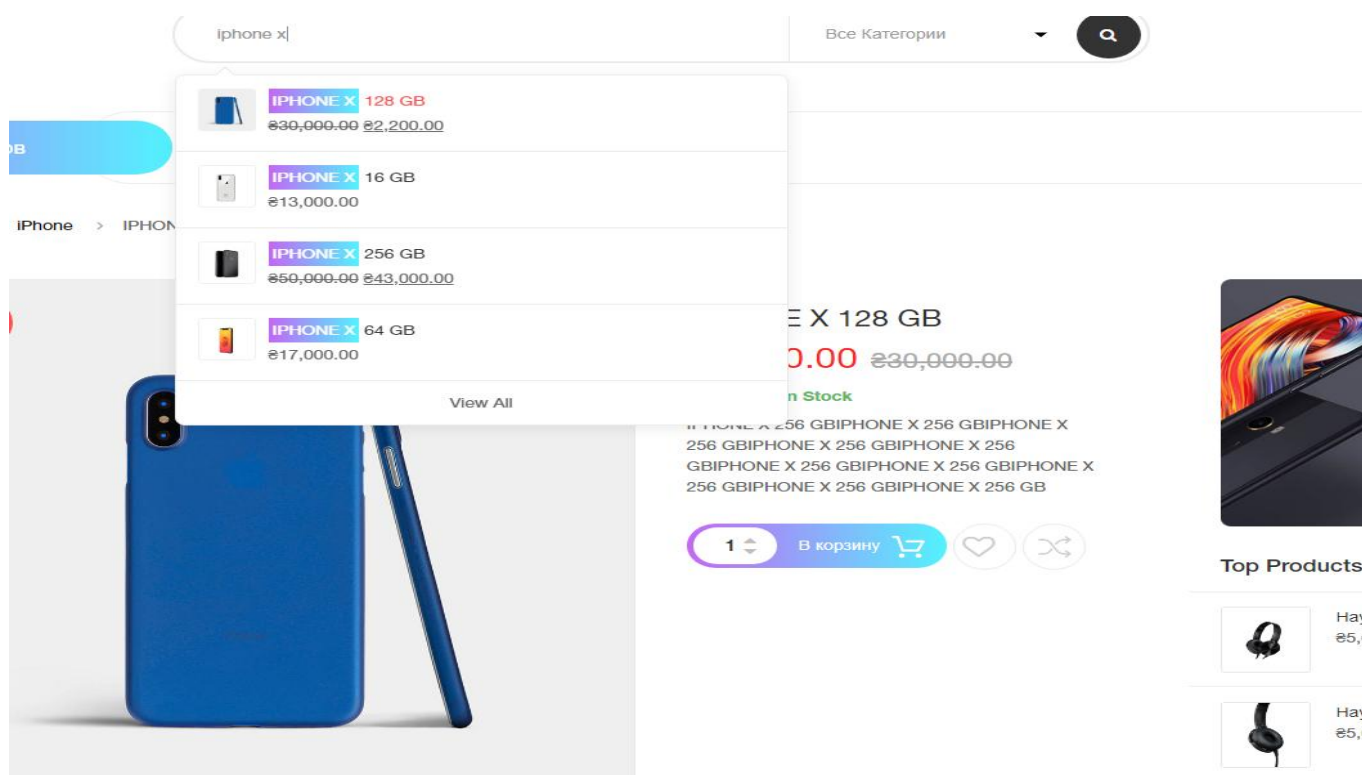


Рис 4.14 – Відображення співпадінь за ключовим словом

Розглянемо наступний розділ сайту на головній сторінці, в якому зображені найпопулярніші товари, та коротко розглянемо дії розробника для отримання такого ефекту, рис 4.15.

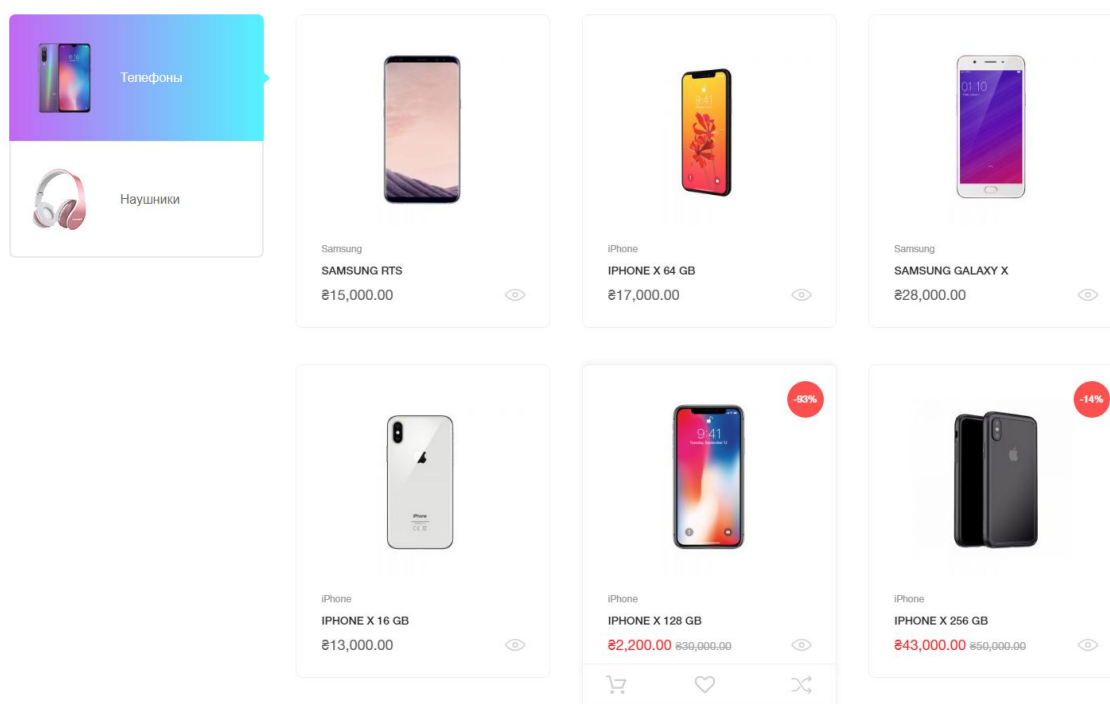


Рис 4.15 – Перегляд товарів за сортуванням «по популярності»

Насамперед необхідно сортувати будь який вид товару в певних категоріях, для цього CMS WP пропонує нам «Wp хуки» якими можна автоматично сортувати позиції товарів, в нашому випадку використовується скрипт зображений на рис 4.16.

```

1 add_filter( 'woocommerce_get_catalog_ordering_args', 'custom_woocommerce_get_catalog_ordering_args' );
2
3 function custom_woocommerce_get_catalog_ordering_args( $args ) {
4     $orderby_value = isset( $_GET['orderby'] ) ? wc_clean( $_GET['orderby'] ) : apply_filters( 'woocommerce_default_catalog_orderby', get_option( 'woocommerce_default_catalog_orderby' ) );
5     if ( 'random_list' == $orderby_value ) {
6         $args['orderby'] = 'most_popular';
7         $args['order'] = '';
8         $args['meta_key'] = '';
9     }
10    return $args;
11 }
12
13 add_filter( 'woocommerce_default_catalog_orderby_options', 'custom_woocommerce_catalog_orderby' );
14 add_filter( 'woocommerce_catalog_orderby', 'custom_woocommerce_catalog_orderby', 100 );
15
16 function custom_woocommerce_catalog_orderby( $sortby ) {
17     $sortby['random_list'] = __( 'Популярные' );
18     return $sortby;
19 }

```

Рис 4.16 – Скрипт сортування товарів

Необхідно зауважити, що розміщення скрипту необхідно робити в файл functions.php, який відповідає за скрипти які не запропоновані за замовчуванням розробниками теми. Також необхідно звернути увагу, функції використовує розробник саме від плагіну Woocommerce, це спрощує написання скрипту, також подібний варіант можна знайти в відкритих бібліотеках WP.

Або написати аях скрипт, який буде працювати швидше, але потребує певних знань та навичок розробника, тому використовується «хуки» від плагіну та самого WP, що спрощує читання коду, навіть для користувачів не знайомих з PHP.

Розглянемо додатково написаний розробником скрипт знижок з відображенням в товарах, як зображено на рис. 4.17.

```

1 add_action( 'woocommerce_cart_calculate_fees','hml_fee_two_plus_one', 10, 1 );
2 function hml_fee_two_plus_one( WC_Cart $cart_object ) {
3     global $wpdb, $woocommerce;
4
5     if ( is_admin() && ! defined( 'DOING_AJAX' ) ) { return; }
6
7     $arr_cart = $cart_object->get_cart();
8     if ( 0 == sizeof($arr_cart) ) { return; }
9     $count = 0;
10    $min_price = 0;
11    $category_id = 914;
12    $cats = hml_get_category_gender_line( $category_id );
13
14    foreach( $arr_cart as $item_key => $value ) {
15        if ( has_term($cats, 'product_cat', $value['product_id'] ) ) {
16            $count += $value['quantity'];
17            $product = $value['data'];
18            $price = $product->get_price();
19            if ( $price ) { $prices[] = floatval($price); }
20        }
21    }
22
23    $min_price = max( $min_price, min($prices) );
24
25    if ( $count >= 2 && $min_price > 0 ) {
26        $fee = -1 * $min_price/2;
27        $cart_object->add_fee( __(' '), $fee );
28    }
29
30 }

```

Рис 4.17 – Скрипт знижок

Частково розглянемо цікаві моменти скрипту :

$\$min_price = \max(\$min_price, \min(\$prices));$

$\text{if } (\$count \geq 2 \ \&\& \ \$min_price > 0) \{$

$\$fee = -1 * \$min_price/2;$

$\$cart_object->add_fee(__(' '), \$fee);$

Змінна $\$min_price$ тримає в собі найбільше значення ціни, яке потім перевіряється на змінну $\$count \geq 2$, тобто якщо кількість товарів більше 2 та $\$min_price > 0$, мінімальна ціна більша за 0, спрацює наступний скрипт. Змінна $\$fee$ містить в собі формулу розрахунку, а потім повертається метод розрахунків з урахуванням нової ціни, тобто знижки.

Розробнику додатково було поставлене завдання створити на базі плагіну «woocommerce» можливість роботи з доставкою товарів за базою відділень пошти від «Нова Пошта», це можливо лише завдяки використанню API від розробників поштової служби, яка дає змогу в режимі 24/7 слідкувати за всіма відділеннями та формувати зручний акт замовлення товару для адміністраторів сайту-магазину, а також є можливість коригувати деякі елементи коду для коректного відображення, демонстрація доставки за базою даних від «Нова Пошта» зображено на рис 4.18.

The screenshot displays a web form for shipping information. On the left, there is a form with fields for 'Country' (set to 'Украина'), 'Phone' (with a red asterisk and value '+38095991122334'), and 'Email' (with a red asterisk). Below these is a section titled 'Укажите адрес доставки' (Specify the delivery address) with a dropdown for 'Region' (set to 'Тернопольская') and a list for 'City' (with 'Горынка' selected). On the right, a summary panel shows 'Подытог' (Subtotal) as 895,000.00, 'Доставка' (Delivery) with radio buttons for 'Доставка службой Новая почта' (selected) and 'Оплата при доставке' (unselected), and 'Итого' (Total) as 895,000.00. At the bottom of the summary panel is a blue button labeled 'Подтвердить заказ' (Confirm order).

Рис 4.18 – Работа API від «Нова Пошта»

Для більшого розуміння, користувачу необхідно обрати область, населений пункт та необхідне відділення нової пошти.

При роботі з API необхідно було внести власні зміни до програмного коду, які розглянемо на рис.4.19.

```
add_filter( 'woocommerce_update_order_review_fragments', 'awoohc_add_update_form_billing', 99 );
function awoohc_add_update_form_billing( $fragments ) {
    $checkout = WC()->checkout();
    ob_start();
    ?>
    <div class="woocommerce-billing-fields__field-wrapper">
        <?php
        $fields = $checkout->get_checkout_fields( 'billing' );
        foreach ( $fields as $key => $field ) {
            if ( isset( $field['country_field'], $fields[ $field['country_field'] ] ) ) {
                $field['country'] = $checkout->get_value( $field['country_field'] );
            }
            woocommerce_form_field( $key, $field, $checkout->get_value( $key ) );
        }
        ?>
    </div>
    <?php
    $art_add_update_form_billing = ob_get_clean();
    $fragments['.woocommerce-billing-fields'] = $art_add_update_form_billing;

    return $fragments;
}
```

Рис 4.19 – Внесення змін до скрипту доставки

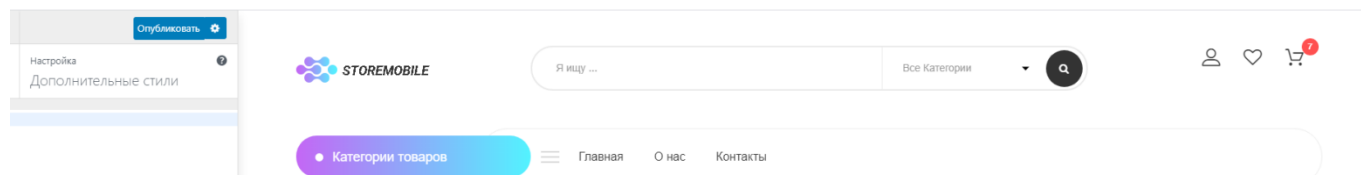
Справа в тому, що в випадку нашої теми необхідно було додати цей фрагмент коду за для коректного відображення інформації на сторінці замовлення товару, іноді користувацькі шаблони з їх стилями та функціями можуть не

працювати з API, або плагінами. Даний фрагмент коду був взятий з архіву бази знань нової пошти, адже ця проблема виникала досить часто в багатьох розробників.

Розглянемо наступний скрипт, який виправив відображення полей для вибору місця доставки. Необхідно в файлі cart-shipping внести зміни до `if (1 < count($available_methods))` на `if (2 > count($available_methods))`

Тобто, змінити перевірку на допустиму кількість полей в методі виводу. Говорити про частину back-end можна безкінечно довго. Адже кожен елемент на сайті так, або інакше розроблений за допомогою PHP та може бути змінений згідно необхідних параметрів для більшої зручності

Звичайно, що для простого користувача на сайті абсолютно не видно всіх функціональних можливостей сайту, а на перший план виходить саме візуальна частина сайту, за допомогою якої навчилися маніпулювати користувачами та притягувати їх погляд на ті товари, які хочуть продати більш за все, тому розглянемо візуальні скрипти CSS та JS, а також роботу з візуальними ефектами, зображено на рис 4.20.



4.20 – Елемент теми без користувацького стилю

В нашому випадку – видаляти особовий кабінет не було ніякого сенсу, можливо він колись буде потрібним для замовника, тому прийнято було рішення внести правки, звертаємо увагу на рис 4.21

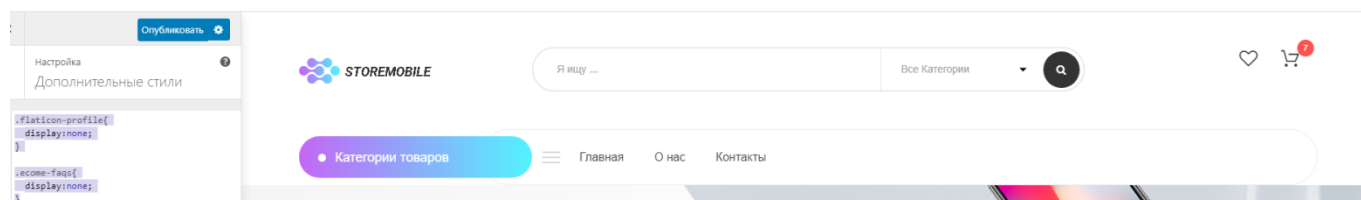


Рис 4.21 – Користувацькі правки стилів

Дуже часто розробники використовують подібні хитрощі, які прибирають візуальні дефекти шаблону який не підпадає до технічного завдання.

Розглянемо привабливий слайдер «карусель», створений за допомогою бібліотеки JQuery, та CSS, також був інтегрований в шаблон для коректного відображення, розглянемо спочатку як виглядає слайдер, рис 4.22.

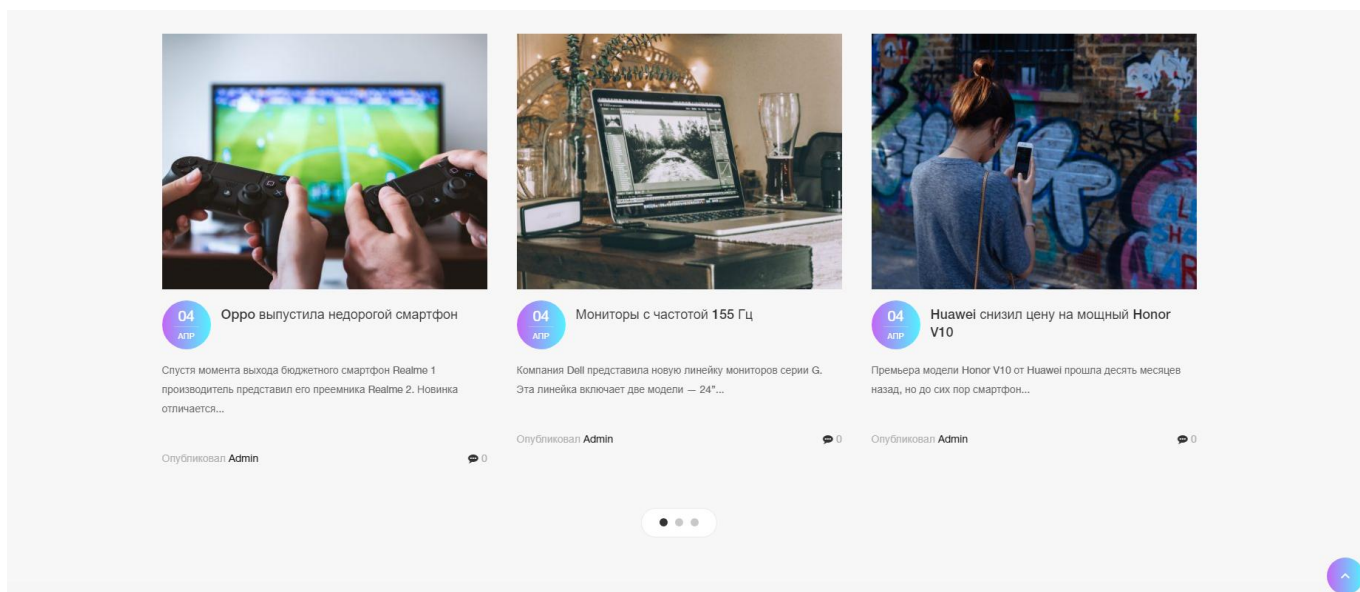


Рис 4.22 – Слайдер «карусель»

Розглянемо більш детально, як було реалізований такий візуальний ефект, скрипт зображено на рис 4.23.

```

1 <div class="carousel"><div class="multiple-items woocommerce products">
2
3 <?php global $post;
4 $args = array(
5     'post_type' => 'product',
6     'posts_per_page' => 10,
7     'columns' => 4,
8     'meta_query' => WC()->query->get_meta_query(),
9     'post_in' => array_merge( array( 0 ), wc_get_product_ids_on_sale() );
10 );
11 $myposts = get_posts( $args );
12 foreach( $myposts as $post ) : setup_postdata($post); ?>
13
14 <div <?php post_class( $classes ); ?>>
15 <?php
16 do_action( 'woocommerce_before_shop_loop_item' );
17 do_action( 'woocommerce_before_shop_loop_item_title' );
18 do_action( 'woocommerce_shop_loop_item_title' );
19 do_action( 'woocommerce_after_shop_loop_item_title' );
20 do_action( 'woocommerce_after_shop_loop_item' );
21 ?>
22 </div>
23
24 <?php endforeach; ?>
25 <?php wp_reset_postdata() ?>
26
27 </div></div>

```

Рис 4.23 – Скрипт слайдеру «карусель»

Нажаль візуально слайдер ще не функціонує, це його back-end сторона, якщо додати трішки CSS одразу все буде на своїх місцях, рис 4.24.

```

30
31 .multiple-items {display: flex !important; flex-wrap: wrap; margin-bottom: 25px;}
32
33 .multiple-items img {
34   -moz-backface-visibility: visible;
35   -webkit-backface-visibility: visible;
36   backface-visibility: visible;
37 }
38
39 .slick-list {margin: 0 -1px 0 0 !important; padding: 0 1px 0 0 !important; width: 100%;}
40
41 .slick-track {border: solid #e9eae; border-width: 0 0 0 1px; display: flex !important; padding-bottom: .1rem;}
42
43 .carousel .product {
44   border: solid #e9eae;
45   border-width: 1px 1px 1px 0;
46   background: #fff;
47   padding: 15px;
48   list-style: none;
49   text-align: center;
50   position: relative;
51   margin-bottom: -.1rem;
52   height: auto;
53 }
54
55 .multiple-items > .product {width: 25%;}

```

Рис 4.24 – Стили для слайдеру «карусель»

Після додавання стилів всі об'єкти набули тієї візуальної форми, що користувачі можуть спостерігати на веб-сайті. Підсумуючи розробку елементів сайту «StoreMobile» можна сказати, що як і в 80% випадків розробники витрачають свій час на вирішення завдань від замовника та на пошук і виправлення недоліків шаблонів тем, з якими необхідно мати справу.

Нажаль, веб-програмування з метою послабити тиск на програмістів та їх час та оптимізувати розробку веб-ресурсів, трішки зійшли з вірного курсу, і сучасні розробники пожинають плоди «оптимізації». Бувають випадки коли замінити лише одну кнопку на ту яка необхідно – це година пошуку саме функції яка відпрацьовує при натисканні. Повноцінна робота з чужим кодом та користувацькими темами має на увазі роботу одразу з БД, PHP, JS, JQuery, Css, html та роботі з великою кількістю плагінів та API, без необхідних знань буде досить важко налаштувати хоча б відображення шаблону на хостингу, не кажучи про якісь доопрацювання.

4.5. Тестування сайту

Сучасне тестування сайту полягає в тому, аби сайт можна було зручно відкривати та використовувати з різних пристроїв, наприклад з мобільних телефонів чи планшетів, або з різних версій браузерів, хоча останній пункт не

завжди можна виконати, адже технології стрімко розвиваються та деякі браузери стають застарілими і неспроможними відтворювати певну інформацію.

То ж протестуємо сайт на прикладі роботи на «iPad» з розмірами екрану 786x1024, як зображено на рис 4.25.

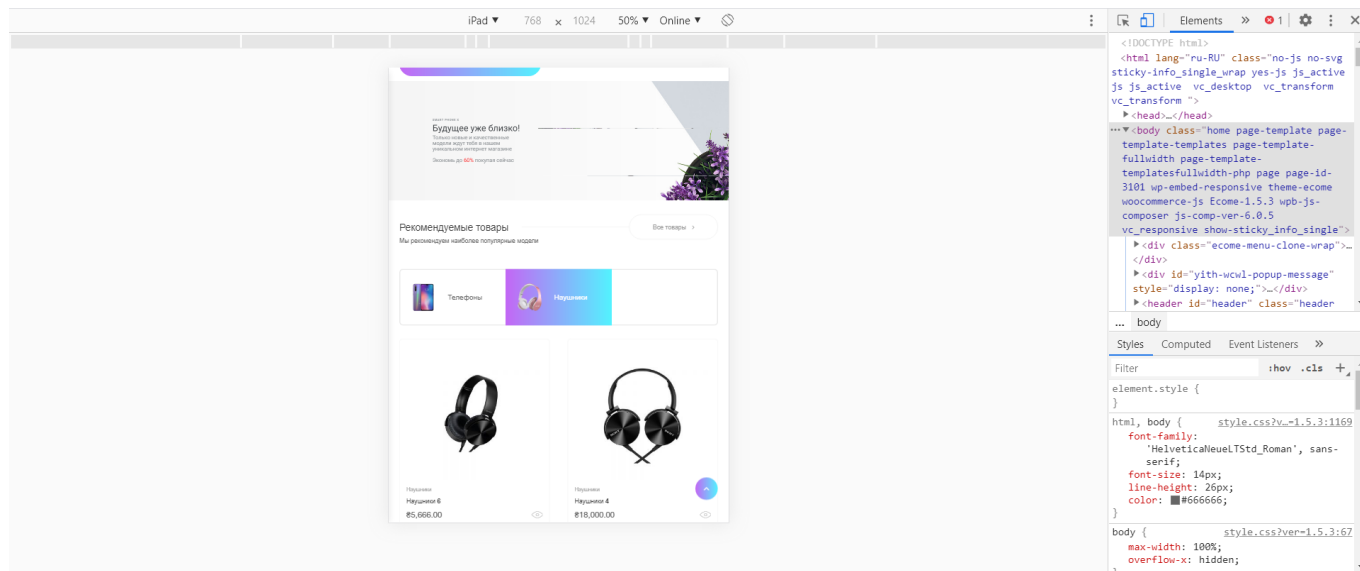


Рис 4.25 □ Відображення сайту на «iPad»

Як ми можемо бачити, сайт відображений коректно згідно технічного завдання та за допомогою технології «Bootstrap», яка дозволяє робити еластичні сайти. Переглянемо відображення інтернет-магазину з браузера «Google Chrome» на рис.4.26.

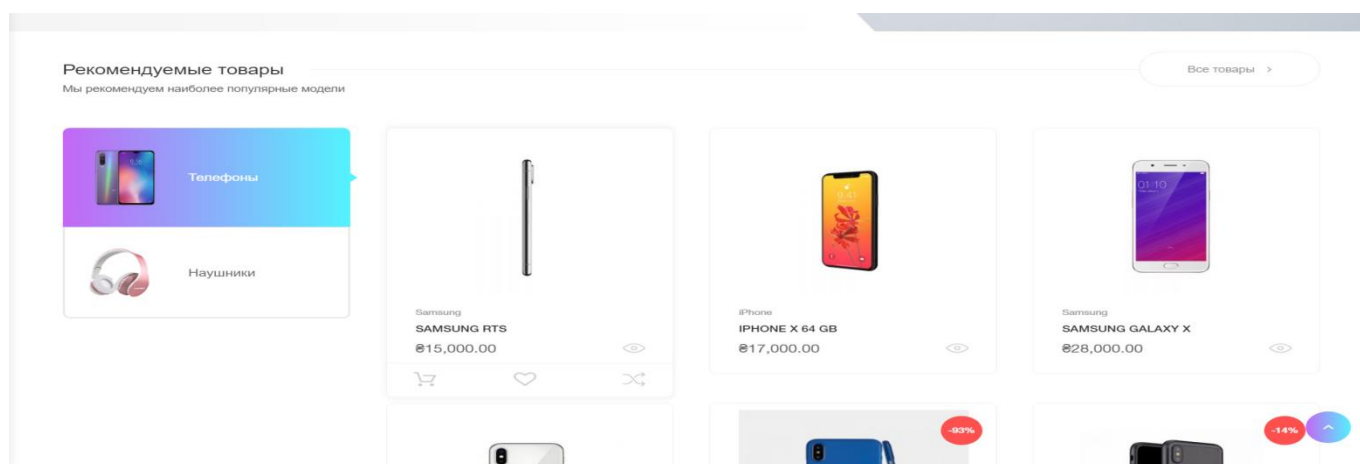


Рис.4.26 □ Відображення інтернет магазину з браузера «Google Chrome»

Отже, поставлені завдання виконано вірно і сайт повністю відповідає потребам замовника.

ВИСНОВКИ

В результаті виконання даної бакалаврської роботи було розроблено інтернет-магазин «StoreMobile». При створенні сайту, були використані сучасні технології, у вигляді системи управління вмістом та різноманітних плагінів. Розроблена фізична та логічна структура сайту в більш спрощеному вигляді. Дотримано всі вимоги технічного завдання та проведено тестування на різних браузерях, та мобільних пристроях.

Всі розробки велись за допомогою скриптових мов програмування, таких як: PHP, Js , Ajax, та різних плагінів. Основними самостійно розробленими елементами сайту, є головна сторінка сайту, форма зв'язку з адміністраторами сайту та створення каталогу товарів і кошика сайту.

Головними порадами в подальшій розробці та використанні сайту, є підключення SSL-сертифікату, який шифрує підключення користувачів та дає змогу проводити навіть банківські операції з захистом даних, та приховувати паролі користувачів. А також, оптимізація роботи PHP скриптів, у вигляді видалення зайвих фрагментів коду від функціоналу, який зовсім не потрібний.

Сайт прийнятий замовником і розташований в мере Internet, є акт впровадження.

СПИСОК ЛІТЕРАТУРИ

1. Вовк О. Б. Методи та засоби підвищення життєздатності веб-сайту як інформаційного продукту: автореф. дис. ... канд. техн. наук: 05.13.06 / Вовк Олена Борисівна; М-во освіти і науки України, Нац. ун-т «Львів. Політехніка». – Львів, 2013. – 20 с.: іл., табл. – Бібліогр.: с. 17-18.
2. Горностаева Е. А. Современные проблемы дизайна сайтов и вызовы нового времени / Горностаева Е. А. // Молодой ученый. — 2015. — №1. — С. 38-40.
3. Дакетт Д. JavaScripte и jQuery. Интерактивная веб-разработка / Д. Дакетт. – Москва: Издательство „Э”, 2017. – 640 с.
4. Дорофеєва О. Удосконалення доступу до галузевого веб-ресурсу / Ольга Дорофеєва // Бібліотечний вісник. – 2010. – № 1. – С. 3-9.
5. Евдокимов Н.С. Создание сайтов / Н.С. Евдокимов. – Санкт-Петербург: Питер, 2014. – 410 с.
6. Інькова Н. А. Створення Web-сайтів : Навчально-методичний посібник (Електронний ресурс) / Інькова Н. А., Зайцева Е. А., Кузьміна Н. В., Толстих С. Г. – Режим доступу до електронного ресурсу: <http://club-edu.tambov.ru/methodic/fio/p5.doc> (дата звернення 26.05.2020).
7. Компанєєтс М.О. Принципи проектування ефективних вебсайтів / М.О. Компанєєтс // Молодий вчений. – 2015. – № 9. – Част. 2. – С. 106-108.
8. Котеров Д.В. PHP 7 / Д.В. Котеров. – Санкт-Петербург: БХВ-Петербург, 2016. – 1088 с. Колисниченко Д.Н. PHP и SQL. Разработка Web-приложений. / Д.Н. Колисниченко. – СПб.: БХВ-Петербург, 2013. – 543 с.
9. Матат Д. Нові технології: веб-квест / Дарина Матат // Освіта України. – 2014. – 23 черв. (№ 23). – С. 10.
10. Мороз Виктор. Защита авторских прав на веб-сайт / Виктор Мороз // Юридическая Практика. – 2010. – № 16(643). – С. 18-19.
11. Основні етапи розробки веб-сайту (Електронний ресурс) / [укл. Юрчак І. Ю.] / Національний університет "Львівська політехніка". – Режим доступу до

електронного ресурсу : <http://www.victoria.lviv.ua/html/wp/ssteps.html> (дата звернення 25.05.2020).

12. Паньчик Б. Розробка ефективних веб-додатків (текст на англ. мові) / Беата Паньчик // Актуальні проблеми економіки. – 2013. – № 11. – С. 443-251.

13. Преимущества web-сайтов на CMS (Електронний ресурс). Режим доступу до електронного ресурсу – <http://webfactura.ru/services/website/advantage/> (дата звернення 25.05.2020).

14. Система управления сайтом (CMS) (Електронний ресурс). Режим доступу до електронного ресурсу – <http://www.grizliart.ru/cms/> (дата звернення 26.05.2020).

15. Тарнавська Т.В. Сутність інформаційних технологій в освіті / Т.В. Тарнавська // – К.: Кондор. – 2013. – 210с

16. Угрин Д. І. Веб-технології та веб-дизайн: [навч. посіб. для студ. вищ. навч. закл.] / Д. І. Угрин, М. І. Маниліч, Б. Є. Деркач. – Чернівці : Золоті литаври, 2012. – 285 с.: іл., табл. – Бібліогр.: с. 284-285.

17. Філінович В. В. Веб-сайт як особливий об'єкт права інтелектуальної власності: автореф. дис. ... канд. юрид. наук: 12.00.03 / Філінович Валерія Вікторівна; М-во освіти і науки України, Київ. нац. ун-т ім. Т. Шевченка. – Київ, 2015. – 19 с. – Бібліогр.: с. 15-17.

ДОДАТОК А.**Програма**